Group Signatures from Lattices: Simpler, Tighter, Shorter, Ring-based

San Ling, Khoa Nguyen, Huaxiong Wang

Division of Mathematical Sciences,
School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore.
{lingsan, khoantt, hxwang}@ntu.edu.sg

Abstract. We introduce a lattice-based group signature scheme that provides several noticeable improvements over the contemporary ones: simpler construction, weaker hardness assumptions, and shorter sizes of keys and signatures. Moreover, our scheme can be transformed into the ring setting, resulting in a scheme based on ideal lattices, in which the public key and signature both have bit-size $\widetilde{\mathcal{O}}(n \cdot \log N)$, for security parameter n, and for group of N users. Towards our goal, we construct a new lattice-based cryptographic tool: a statistical zero-knowledge argument of knowledge of a valid message-signature pair for Boyen's signature scheme (Boyen, PKC'10), which potentially can be used as the building block to design various privacy-enhancing cryptographic constructions.

1 Introduction

Group signatures [CvH91] have been an active research topic in public-key cryptography. Such schemes allow users of a group to anonymously sign messages on behalf of the whole group (*anonymity*). On the other hand, in cases of disputes, there is a tracing mechanism which can link a given signature to the identity of the misbehaving user (*traceability*). These two appealing features allow group signatures to find applications in various real-life scenarios, such as digital right management, anonymous online communications, e-commerce systems, and much more. On the theoretical front, designing secure and efficient group signature schemes is interesting and challenging, since those advanced constructions usually require a sophisticated combination of carefully chosen cryptographic ingredients: digital signatures, encryptions, and zero-knowledge protocols. Over the last two decades, numerous group signature schemes have been proposed (e.g., [CS97,ACJT00,BMW03,BBS04,BS04,Gro07,LPY12]).

In recent years, lattice-based cryptography, possessing nice features such as provable security under worst-case hardness assumptions, conjectured resistance against quantum computers and asymptotic efficiency, has become one of the most trendy research directions, especially after the emergence of fully-homomorphic encryption schemes from lattices, pioneered by Gentry [Gen09]. Along with other primitives, lattice-based group signatures has received noticeable attention. Prior to our work, several schemes were proposed, each of which has its own strengths and weaknesses. The first group signature from lattices was introduced by Gordon et al. [GKV10]. While their scheme is of great theoretical interest, its public key and signature have sizes $N \cdot \widetilde{\mathcal{O}}(n^2)$, for security parameter n, and for group of N users. In terms of efficiency, this is a noticeable disadvantage when the group is large, e.g., group of all employees of a big company. Camenisch et al. [CNR12] later proposed lattice-based anonymous attribute tokens system - a generalization of group signature. Their scheme supports CCA-anonymity, a stronger security requirement than the relaxed notion CPA-anonymity achieved by [GKV10], but the signature size is still linear in N. The linear-size barrier was finally overcome by Laguillaumie et al. [LLLS13], who designed a scheme featuring public key and signature sizes $\log N \cdot \widetilde{\mathcal{O}}(n^2)$. Yet, their scheme requires large parameters (e.g., $q = \log N \cdot \widetilde{\mathcal{O}}(n^8)$), and its anonymity and traceability properties have to rely on the hardness of $\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^8)}$ and $\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^{7.5})}$, respectively. Thus, the scheme produces significant overheads in terms of hardness assumptions, considering the fact that it is constructed

based on Boyen's signature [Boy10] and the Dual-Regev encryption [GPV08] which rely on much weaker assumptions. Recently, Langlois et al. [LLNW14] introduced a lattice-based group signature scheme with verifier-local revocation, that also achieves logarithmic signature size. However, their scheme only satisfies a weak security model suggested by Boneh et al. [BBS04]. As in the schemes from [GKV10,CNR12,LLLS13], we consider the currently strongest model for static groups provided by Bellare et al. [BMW03].

The present state of lattice-based group signatures raises several interesting open questions. One of them is whether it is possible to design a scheme in the BMW model that simultaneously achieves signature size $\log N \cdot \widetilde{\mathcal{O}}(n)$ and weak hardness assumptions. Another open question, pointed out in [LLLS13], is to construct group signatures based on the ring variants of the Small Integer Solutions (SIS) and Learning with Errors (LWE) problems. This would make a noticeable step towards practice, since in those schemes, the public key size can be as small as $\log N \cdot \widetilde{\mathcal{O}}(n)$. Furthermore, we remark that the design approach of [GKV10,CNR12,LLLS13] are relatively complex. First, in all of these schemes, the encryption layer (needed for enabling traceability) has to be initialized in accordance with the signature layer (used for key generation), which, to some extent, limits the choice of encryption mechanisms. In addition, the encryption layer requires the costly generation of at least $\mathcal{O}(\log N)$ matrices in $\mathbb{Z}_q^{n \times m}$, and the signer has to encrypt at least $\log N \cdot \widetilde{\mathcal{O}}(n)$ bits, which leads to a growth in public key and signature sizes. Moreover, these schemes have to employ involved zero-knowledge protocols to prove the well-formedness of the obtained ciphertexts: in [GKV10,CNR12], the main protocols are obtained by OR-ing N proofs, while in [LLLS13], $\log N + 2$ different proofs are needed. This somewhat unsatisfactory situation highlights the challenge of simplifying the design of lattice-based group signatures.

Our Contributions and Summary of Our Techniques.

In this work, we reply positively to all the open questions discussed above. Specifically, we introduce a lattice-based group signature scheme in the random oracle model (in Section 4), which simultaneously achieves the following features:

- The public key and signature have sizes $\log N \cdot \widetilde{\mathcal{O}}(n^2)$ and $\log N \cdot \widetilde{\mathcal{O}}(n)$, respectively [1]. In comparison with [LLLS13], the key is around 4 times smaller, and the signature contains a shorter ciphertext.
- The scheme relies on relatively weak hardness assumptions: it is CCA-anonymous and traceable if $\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^2)}$ is hard in the worst-case. In contrast to [LLLS13], the scheme produces no overhead in terms of security: its anonymity and traceability properties rely exactly on the hardness assumptions of the underlying encryption scheme and signature scheme, respectively.

Furthermore, our scheme can be transformed into the ring setting, resulting in a scheme based on ideal lattices (in Section 5), in which the key and signature both have size $\widetilde{\mathcal{O}}(n \cdot \log N)$. In Table 1, we summarize the features of our two schemes in comparison with the existing ones.
Another contribution of this work is that our schemes are obtained via a simple design approach. We rely on Boyen's signature scheme [Boy10], and consider group of $N = 2^\ell$ users, where each user is identified by a string $d \in \{0,1\}^\ell$, as in [LLLS13]. Yet, in our scheme, the user's secret key is simply a Boyen signature $\mathbf{z} \in \mathbb{Z}^{2m}$ on d (in [LLLS13], it is a matrix in $\mathbb{Z}^{2m \times 2m}$ - which is $2m = \widetilde{\mathcal{O}}(n)$ times longer). To sign a message on behalf of the group, the user first encrypts his identity d to obtain a ciphertext $\mathbf{c}$, and then generates a zero-knowledge argument to prove that he possesses a valid message-signature pair $(d, \mathbf{z})$ for Boyen's signature scheme, and that $\mathbf{c}$ is a

[1] It was noted by Bellare et al. [BMW03], that the dependency of keys and signatures sizes on $\log N$ is unavoidable for group signature schemes in the their model.

Scheme	[GKV10]	[CNR12]	[LLLS13]	Section 4	Section 5
Signature size	$N \cdot \widetilde{\mathcal{O}}(n^2)$	$N \cdot \widetilde{\mathcal{O}}(n^2)$	$\log N \cdot \widetilde{\mathcal{O}}(n)$	$\log N \cdot \widetilde{\mathcal{O}}(n)$	$\log N \cdot \widetilde{\mathcal{O}}(n)$
Public key size	$N \cdot \widetilde{\mathcal{O}}(n^2)$	$N \cdot \widetilde{\mathcal{O}}(n^2)$	$\log N \cdot \widetilde{\mathcal{O}}(n^2)$	$\log N \cdot \widetilde{\mathcal{O}}(n^2)$	$\log N \cdot \widetilde{\mathcal{O}}(n)$
Anonymity	$\mathsf{SIVP}_{\widetilde{\mathcal{O}}(n^2)}$	$\mathsf{SIVP}_{\widetilde{\mathcal{O}}(n^2)}$	$\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^8)}$	$\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^2)}$	$\mathsf{SVP}^{\infty}_{\log N \cdot \widetilde{\mathcal{O}}(n^{3.5})}$
Traceability	$\mathsf{SIVP}_{\widetilde{\mathcal{O}}(n^{1.5})}$	$\mathsf{SIVP}_{\widetilde{\mathcal{O}}(n^2)}$	$\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^{7.5})}$	$\mathsf{SIVP}_{\log N \cdot \widetilde{\mathcal{O}}(n^2)}$	$\mathsf{SVP}^{\infty}_{\log N \cdot \widetilde{\mathcal{O}}(n^2)}$

Table 1. Comparison among lattice-based group signature schemes, for security parameter n, and groups of N users. The [GKV10] scheme and our scheme in Section 5 only satisfy the CPA-anonymity notion, while the schemes from [CNR12] and [LLLS13], and our scheme in Section 4 support the stronger notion CCA-anonymity.

correct encryption of d. The protocol then is repeated to make the soundness error negligibly small, and then is made non-interactive using the Fiat-Shamir heuristic. The group signature is simply the pair $(\mathbf{c}, \Pi)$, where Π is the obtained non-interactive argument. To verify a signature, one checks Π, and to open it, the group manager decrypts $\mathbf{c}$. We remark that in our design, the signer has to encrypt only $\ell = \log N$ bits. Furthermore, the underlying encryption scheme is totally independent of the underlying standard signature (i.e., Boyen's signature in this case). This provides us a flexible choice of encryption schemes.

1. In the scheme in Section 4, to achieve CCA-anonymity, we rely on a CCA-secure encryption scheme, obtained by the standard technique of combining a one-time signature scheme and an identity-based encryption (IBE) scheme [BCHK07]. In particular, we employ the IBE scheme by Gentry et al. [GPV08] to gain efficiency in the random oracle model.
2. In the ring-based scheme in Section 5, since our main goal is efficiency, we employ the CPA-secure encryption scheme from [LPR13], for which the public key and ciphertext consist of only 2 ring elements.

In the process, we introduce a new lattice-based cryptographic tool: a statistical zero-knowledge argument of knowledge of a valid message-signature pair for Boyen's signature scheme. We remark that previous protocols in lattice-based cryptography (e.g., [MV03][Lyu08][LNSW13]) only allow to prove in zero-knowledge the possession of a signature on a *publicly given* message. The challenging part is to hide *both* the signature and message from the verifier, which we overcome by a non-trivial technique described in Section 3. We believe that our new protocol is of independent interest. Indeed, apart from group signatures, such protocols are essential for designing various privacy-enhancing constructions, such as anonymous credentials [CL01], compact e-cash [CHL05], policy-based signatures [BF14], and much more.

Comparison to related work. In a concurrent and independent work, Nguyen, Zhang and Zhang [NZZ15], based on a new zero-knowledge protocol corresponding to a simple identity-encoding function, also obtain a simpler lattice-based group signature than [GKV10,LLLS13]. In the [NZZ15] scheme, the public key size and signature size are shorter by a $\mathcal{O}(\log N)$ factor than in the previous works, and are shorter than ours. On the other hand, the user's secret key in [NZZ15] is still a matrix in $\mathbb{Z}^{2m \times 2m}$ (as in [LLLS13]), and the scheme requires larger parameters, e.g., $q = m^{2.5} \max(m^6 \omega(\log^{2.5} m), 4N)$, as well as stronger security assumptions than ours.

2 Preliminaries

NOTATIONS. For integer $n \geq 1$, we denote by $[n]$ the set $\{1, \ldots, n\}$. The set of all permutations of k elements is denoted by $\mathcal{S}_k$. We assume that all vectors are column vectors. The concatenation

of vectors $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^k$ is denoted by $(\mathbf{x}\|\mathbf{y})$. We denote the column concatenation of matrices $\mathbf{A} \in \mathbb{R}^{n\times m}$ and $\mathbf{B} \in \mathbb{R}^{n\times k}$ by $\big[\mathbf{A}\big|\mathbf{B}\big]$. The identity matrix of order k is denoted by $\mathbf{I}_k$. If S is a finite set, $y \xleftarrow{\$} \mathrm{S}$ means that y is chosen uniformly at random from S.

2.1 Group Signatures

Definition 1 ([BMW03]). A group signature scheme is a tuple of 4 polynomial-time algorithms:

- KeyGen: This randomized algorithm takes as input $1^n, 1^N$, where $n \in \mathbb{N}$ is the security parameter and $N \in \mathbb{N}$ is the number of group users, and outputs a triple $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk})$, where gpk is the group public key; gmsk is the group manager's secret key; and $\mathsf{gsk} = \{\mathsf{gsk}[i]\}_{i\in\{0,\ldots,N-1\}}$, where for $i \in \{0,\ldots,N-1\}$, $\mathsf{gsk}[i]$ is the secret key for the group user of index i.
- Sign: This randomized algorithm takes as input a secret signing key $\mathsf{gsk}[i]$ for some $i \in \{0,\ldots,N-1\}$, and a message M, and returns a group signature Σ on M.
- Verify: This deterministic algorithm takes as input the group public key gpk, a message M, a purported signature Σ on M, and returns either 1 (Valid) or 0 ($\mathsf{Invalid}$).
- Open: This deterministic algorithm takes as input the group manager's secret key gmsk, a message M, a signature Σ on M, and returns an index $i \in \{0,\ldots,N-1\}$, or $\perp$ (to indicate failure).

Correctness. The correctness requirement for a group signature scheme is as follows. For all $n, N \in \mathbb{N}$, all $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk})$ produced by $\mathsf{KeyGen}(1^n, 1^N)$, all $i \in \{0,\ldots,N-1\}$, and all $M \in \{0,1\}^*$,

$$\mathsf{Verify}\big(\mathsf{gpk}, M, \mathsf{Sign}(\mathsf{gsk}[i], M)\big) = 1 \quad \text{and} \quad \mathsf{Open}\big(\mathsf{gmsk}, M, \mathsf{Sign}(\mathsf{gsk}[i], M)\big) = i.$$

Security Notions. A secure group signature scheme must satisfy two security notions:

- *Traceability* requires that all signatures, even those produced by a coalition of group users and the group manager, can be traced back to a member of the coalition.
- *Anonymity* requires that, signatures generated by two distinct group users are computationally indistinguishable to an adversary who knows all the user secret keys. In Bellare et al.'s model [BMW03], the anonymity adversary is granted access to an opening oracle (CCA-anonymity), namely, it is allowed to see the results of openings of all signatures (except for the target one). Boneh et al. [BBS04] later proposed a relaxed notion, where the adversary cannot query the opening oracle (CPA-anonymity).

Formal definitions of the above notions are provided in Appendix A.

2.2 Average-case Lattices Problems and Their Ring Variants

We first recall the definitions and hardness results for average-case problems SIS, LWE.

Definition 2 ([Ajt96,GPV08]). The $\mathsf{SIS}^p_{n,m,q,\beta}$ problem is as follows: Given uniformly random matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, find a non-zero vector $\mathbf{x} \in \mathbb{Z}^m$ such that $\|\mathbf{x}\|_p \le \beta$ and $\mathbf{A}\mathbf{x} = \mathbf{0} \bmod q$.

If $m, \beta = \mathsf{poly}(n)$, and $q > \sqrt{n}\beta$, then the $\mathsf{SIS}^\infty_{n,m,q,\beta}$ problem (in the ℓ_∞ norm) is at least as hard as SIVP_γ for some $\gamma = \beta \cdot \widetilde{O}(\sqrt{nm})$ (see [GPV08,MP13]).

Definition 3 ([Reg05]). Let $n, m \geq 1$, $q \geq 2$, and let χ be a probability distribution on $\mathbb{Z}$. For $\mathbf{s} \in \mathbb{Z}_q^n$, let $A_{\mathbf{s},\chi}$ be the distribution obtained by sampling $\mathbf{a} \xleftarrow{\$} \mathbb{Z}_q^n$ and $e \hookleftarrow \chi$, and outputting the pair $(\mathbf{a}, \mathbf{a}^T \cdot \mathbf{s} + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$. The $\mathsf{LWE}_{n,q,\chi}$ problem asks to distinguish m samples chosen according to $\mathcal{A}_{\mathbf{s},\chi}$ (for $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$) and m samples chosen according to the uniform distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

If q is a prime power, $b \geq \sqrt{n}\omega(\log n)$, $\gamma = \widetilde{\mathcal{O}}(nq/b)$, then there exists an efficient sampleable b-bounded distribution χ (i.e., χ outputs samples with norm at most b with overwhelming probability) such that $\mathsf{LWE}_{n,q,\chi}$ is as least as hard as SIVP_γ (see [Reg05,Pei09,MM11,MP12]).

We now recall the ring variants of the SIS and LWE, as well as their hardness results. Let $f = x^n + 1$, where n is a power of 2, and let $q > 2$ be prime. Let $\mathsf{R} = \mathbb{Z}[x]/\langle f \rangle$ and $\mathsf{R}_q = \mathsf{R}/q\mathsf{R}$. (As an additive group, $\mathcal{R}_q$ is isomorphic to $\mathbb{Z}_q^n$.) For an element $a = c_0 + c_1 x + \ldots + c_{n-1} x^{n-1} \in \mathsf{R}$, we define $\|a\|_\infty = \max_i(|c_i|)$. For a vector $\mathbf{a} = (a_1, \ldots, a_m) \in \mathsf{R}^m$, we define $\|\mathbf{a}\|_\infty = \max_j(\|a_j\|_\infty)$. To avoid ambiguity, we will denote the multiplication operation of two ring elements by the symbol $\otimes$.

Definition 4 ([LM06,PR06,LMPR08]). The $\mathsf{Ring\text{-}SIS}_{n,m,q,\beta}$ problem is as follows: Given a uniformly random $\mathbf{a} = (a_1, \ldots, a_m) \in \mathsf{R}_q^m$, find a non-zero vector $\mathbf{x} = (x_1, \ldots, x_m) \in \mathsf{R}_q^m$ such that $\|\mathbf{a}\|_\infty \leq \beta$ and $\mathbf{ax} = a_1 \otimes x_1 + \ldots a_m \otimes x_m = 0 \bmod q$.

For $m > \frac{\log q}{\log(2\beta)}$, $\gamma = 16\beta mn \log^2 n$, and $q \geq \frac{\gamma\sqrt{n}}{4 \log n}$, the $\mathsf{Ring\text{-}SIS}_{n,m,q,\beta}$ problem is at least as hard as $\mathsf{SVP}_\gamma^\infty$ in any ideal in the ring R (see, e.g., [LM06]).

Definition 5 ([LPR10]). Let $n, m \geq 1$, $q \geq 2$, and let χ be a probability distribution on $\mathcal{R}$. For $s \in \mathcal{R}_q$, let $A_{s,\chi}$ be the distribution obtained by sampling $a \xleftarrow{\$} \mathcal{R}_q$ and $e \hookleftarrow \chi$, and outputting the pair $(a, a \otimes s + e) \in \mathcal{R}_q \times \mathcal{R}_q$. The $\mathsf{Ring\text{-}LWE}_{n,m,q,\chi}$ problem asks to distinguish m samples chosen according to $\mathcal{A}_{s,\chi}$ (for $s \xleftarrow{\$} \mathcal{R}_q$) and m samples chosen according to the uniform distribution over $\mathcal{R}_q \times \mathcal{R}_q$.

Let $q = 1 \bmod 2n$, $b \geq \omega(\sqrt{n \log n})$ and $\gamma = n^2(q/b)(nm/\log(nm))^{1/4}$. Then there exists an efficient sampleable b-bounded distribution χ such that the $\mathsf{Ring\text{-}LWE}_{n,m,q,\chi}$ problem is at least as hard as $\mathsf{SVP}_\gamma^\infty$ in any ideal in the ring R (see [LPR10]).

Note that the hardness of LWE is not affected if the secret $\mathbf{s}$ is sampled from the error distribution χ [ACPS09]. The same holds for $\mathsf{Ring\text{-}LWE}$ (see [LPR13]). This is called the "Hermite Normal Form" (HNF) of these problems.

2.3 Boyen's "Lattice-mixing" Signature Scheme and Its Ring-based Variant

Boyen's signature scheme [Boy10] is a lattice analogue of Water's pairing-based signature [Wat05]. Here we consider its improved version provided in [MP12]. The scheme uses the following integer parameters: n is the security parameter, ℓ is the message length, $q = \mathsf{poly}(n)$ is sufficiently large, $m \geq 2n \log q$, $\sigma = \Omega(\sqrt{\ell n \log q} \log n)$ and $\beta = \sigma\omega(\sqrt{\log m})$. The public key is a tuple $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$, and the signing key is a trapdoor $\mathbf{T_A}$, where:

- Matrix $\mathbf{A}$ is statistically close to uniform over $\mathbb{Z}_q^{n \times m}$ and its trapdoor $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$ is a short basis for the lattice $\Lambda^\perp(\mathbf{A}) = \left\{\mathbf{x} \in \mathbb{Z}^m : \ \mathbf{A} \cdot \mathbf{x} = \mathbf{0} \bmod q\right\}$. The pair $(\mathbf{A}, \mathbf{T_A})$ is generated by a PPT algorithm $\mathsf{GenTrap}(n, m, q)$ (see [GPV08,AP11,MP12]).
- Matrices $\mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n \times m}$ and vector $\mathbf{u} \in \mathbb{Z}_q^n$ are uniformly random.

To sign a message $d = (d_1, \ldots, d_\ell)$, the signer forms $\mathbf{A}_{(d)} = \big[\mathbf{A} \,|\, \mathbf{A}_0 + \sum_{i=1}^{\ell} d_i \mathbf{A}_i\big] \in \mathbb{Z}_q^{n \times 2m}$, then runs the deterministic algorithm $\mathsf{ExtBasis}(\mathbf{T}_{\mathbf{A}}, \mathbf{A}_{(d)})$ from [CHKP10] to obtain a short basis $\mathbf{T}_{(d)}$ for the lattice $\Lambda^{\perp}(\mathbf{A}_{(d)})$. Finally he runs the PPT algorithm $\mathsf{SamplePre}(\mathbf{T}_{(d)}, \mathbf{A}_{(d)}, \mathbf{u}, \sigma)$ from [GPV08] to output a signature $\mathbf{z} \in \mathbb{Z}^{2m}$ satisfying $\|\mathbf{z}\|_\infty \leq \beta$ and $\mathbf{A}_{(d)}\mathbf{z} = \mathbf{u} \bmod q$. It follows from the improved security reduction in [MP12] that scheme is unforgeable under adaptive chosen-message attack if the $\mathsf{SIS}^{\infty}_{n,m,q,\beta'}$ problem is hard for some $\beta' = \ell\widetilde{\mathcal{O}}(n)$. Therefore, for the given parameters, the security of the scheme can be based on the worst-case hardness of $\mathsf{SIVP}_{\ell\widetilde{\mathcal{O}}(n^2)}$.

The public key in Boyen's signature scheme has bit-size $\ell\mathcal{O}(nm \log q) = \ell\widetilde{\mathcal{O}}(n^2)$, but can be reduced to $\ell\widetilde{\mathcal{O}}(n)$ by transforming the scheme into the ring setting, because the parameter m then can be set as $m = \Omega(\log q)$. This can be done rather straightforwardly, thanks to the constructions of the algorithms $\mathsf{GenTrap}$, $\mathsf{SamplePre}$, and $\mathsf{ExtBasis}$ for ideal lattices given by Stehlé et al. [SSTX09]. For an element $a \in \mathcal{R}_q$, define $\mathsf{rot}(a) \in \mathbb{Z}_q^{n \times n}$ as the matrix whose i-th column is $x^i \otimes a$, for $i = 0, \ldots, n-1$. For a vector $\mathbf{a} = (a_1, \ldots, a_m) \in \mathcal{R}_q^m$, define $\mathsf{rot}(\mathbf{a}) = \big[\mathsf{rot}(a_1) \,|\, \ldots \,|\, \mathsf{rot}(a_m)\big] \in \mathbb{Z}_q^{n \times nm}$.

In the ring variant of Boyen's signature, the public key is a tuple $(\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u) \in \big(\mathcal{R}_q^m\big)^{\ell+2} \times \mathcal{R}_q$, and the signing key is a trapdoor $\mathbf{T}_{\mathbf{a}} \in \mathbb{Z}^{nm \times nm}$ for the lattice $\Lambda^{\perp}\big(\mathsf{rot}(\mathbf{a})\big)$. Similarly, a signature on message $d \in \{0,1\}^\ell$ is a small-norm vector $\mathbf{z} \in \mathcal{R}^{2m}$ such that $\big[\mathbf{a} \,|\, \mathbf{a}_0 + \sum_{i=1}^{\ell} d_i \mathbf{a}_i\big]\mathbf{z} = u \bmod q$. By adapting the security reduction from [MP12] into the ring setting, the security of the scheme can be based on the average-case hardness of $\mathsf{Ring\text{-}SIS}_{n,m,q,\beta'}$ problem for some $\beta' = \ell\widetilde{\mathcal{O}}(n)$, which in turn can be based on the worst-case hardness of $\mathsf{SVP}^{\infty}_{\ell\widetilde{\mathcal{O}}(n^2)}$ on ideal lattices.

2.4 Zero-knowledge Argument Systems for Lattices

We will work with statistical zero-knowledge argument systems, namely, interactive protocols where the soundness property only holds for *computationally bounded* cheating provers, while the zero-knowledge property holds against *any* cheating verifier. More formally, let the set of statements-witnesses $R = \{(y, w)\} \in \{0,1\}^* \times \{0,1\}^*$ be an NP relation. A two-party game $\langle P, V \rangle$ is called an interactive argument system for the relation R with soundness error e if the following two conditions hold:

- $\mathsf{Completeness}$. If $(y, w) \in R$ then $\Pr\big[\langle P(y, w), V(y) \rangle = 1\big] = 1$.
- $\mathsf{Soundness}$. If $(y, w) \notin R$, then for every PPT P^*: $\Pr[\langle P^*(y, w), V(y) \rangle = 1] \leq e$.

An interactive argument system is called statistical zero-knowledge if for any $V^*(y)$, there exists a PPT simulator $\mathcal{S}(y)$ producing a simulated transcript that is statistically close to the one of the real interaction between $P(y, w)$ and $V^*(y)$. A related notion is argument of knowledge, which requires the witness-extended emulation property. For protocols consisting of 3 moves (i.e., commitment-challenge-response), witness-extended emulation is implied by *special soundness* [Gro04], where the latter assumes that there exists a PPT extractor which takes as input a set of valid transcripts with respect to all possible values of the 'challenge' to the same 'commitment', and outputs w' such that $(y, w') \in R$.

Statistical zero-knowledge arguments of knowledge (sZKAoK) are usually constructed using a statistically hiding and computationally binding string commitment scheme. Kawachi et al. [KTX08] designed such commitment scheme from lattices, where the binding property relies on the hardness of $\mathsf{SIVP}_{\widetilde{\mathcal{O}}(n)}$. Using this primitive, Ling et al. [LNSW13] proposed a Stern-type [Ste96] sZKAoK for the $\mathsf{Inhomogeneous\ SIS}$ relation:

$$\mathrm{R}_{\mathsf{ISIS}}(n, m, q, \beta) = \Big\{ \big((\mathbf{A} \in \mathbb{Z}_q^{n \times m}; \mathbf{u} \in \mathbb{Z}_q^n), \mathbf{x} \in \mathbb{Z}^m \big) : \|\mathbf{x}\|_\infty \leq \beta \wedge \mathbf{A}\mathbf{x} = \mathbf{u} \bmod q \Big\}.$$

The core technique in Ling et al.'s work is called Decomposition-Extension. This technique is as follows. Letting $p = \lfloor \log \beta \rfloor + 1$, Ling et al. observe that an integer $x \in [0, \beta]$ if and only if there exist $x_1, \ldots, x_p \in \{0, 1\}$ such that $x = \sum_{j=1}^{p} \beta_j x_j$, where the sequence of integers $\beta_1, \ldots, \beta_p$ is determined as follows:

$$\beta_1 = \lceil \beta/2 \rceil; \beta_2 = \lceil (\beta - \beta_1)/2 \rceil; \beta_3 = \lceil (\beta - \beta_1 - \beta_2)/2 \rceil; \ldots; \beta_p = 1.^2$$

The above observation allows the prover to efficiently decompose $\mathbf{x} \in [-\beta; \beta]^m$ into $\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_p \in \{-1, 0, 1\}^m$ such that $\sum_{j=1}^{p} \beta_j \tilde{\mathbf{x}}_j = \mathbf{x}$. To argue the possession of the $\tilde{\mathbf{x}}_j$'s in zero-knowledge, the prover extends $\tilde{\mathbf{x}}_j$ to $\mathbf{x}_j \in \mathsf{B}_{3m}$, where B_{3m} is the set of all vectors in $\{-1, 0, 1\}^{3m}$ having exactly m coordinates equal 0, m coordinates equal to 1, and m coordinates equal to -1. This set has a helpful property: if π is a permutation of $3m$ elements, then $\mathbf{x}_j \in \mathsf{B}_{3m}$ if and only if $\pi(\mathsf{x}_j) \in \mathsf{B}_{3m}$. Then in the framework of Stern's 3-move protocol, the prover is able to demonstrate that:

1. For each j, a random permutation of $\mathbf{x}_j$ belongs to B_{3m}, which implies that $\mathbf{x}_j \in \mathsf{B}_{3m}$, and thus, $\tilde{\mathbf{x}}_j \in \{-1, 0, 1\}^m$. This will convinces the verifier that $\mathbf{x} \in [-\beta, \beta]^m$.
2. $\mathbf{A}^* \sum_{j=1}^{p} \beta_j (\mathbf{x}_j + \mathbf{r}_j) - \mathbf{u} = \mathbf{A}^* \sum_{j=1}^{p} \beta_j \mathbf{r}_j \bmod q$, where $\mathbf{A}^* \in \mathbb{Z}_q^{n \times 3m}$ is the extended matrix obtained by appending $2m$ "dummy" zero-columns to $\mathbf{A}$, and $\mathbf{r}_1, \ldots, \mathbf{r}_p \in \mathbb{Z}_q^{3m}$ are uniformly "masking" vectors for the $\mathbf{x}_j$'s. This equation implies that $\mathbf{A}\mathbf{x} = \mathbf{A}^* \sum_{j=1}^{p} \beta_j \mathbf{x}_j = \mathbf{u} \bmod q$.

3 New Zero-knowledge Protocols for Lattice-based Cryptography

In this section, we first present a sZKAoK of a valid message-signature pair $(d, \mathbf{z})$ for Boyen's signature scheme ([Boy10], see also Section 2.3). Then we provide a lattice-based verifiable encryption protocol to show that a given ciphertext correctly encrypts d. The combined protocol of these two ones, which will serve as the building block in both constructions of our group signatures, is described in detail in Section 3.3.

3.1 ZKAoK of a Valid Message-Signature Pair for Boyen's Signature Scheme

Suppose that the verification key for Boyen's signature scheme is a tuple $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$. Our goal is to design a sZKAoK of a pair $(d, \mathbf{z}) \in \{0, 1\}^\ell \times \mathbb{Z}^{2m}$ satisfying $\|\mathbf{z}\|_\infty \leq \beta$ and $\mathbf{A}_{(d)}\mathbf{z} = \mathbf{u} \bmod q$, where $\mathbf{A}_{(d)} = \left[\, \mathbf{A} \,|\, \mathbf{A}_0 + \sum_{i=1}^{\ell} d_i \mathbf{A}_i \,\right] \in \mathbb{Z}_q^{n \times 2m}$. We first observe that obtaining a ZKAoK of a Boyen signature on a *given* message d is relatively straightforward: one can just run a zero-knowledge protocol for an $\mathsf{Inhomogeneous\ SIS}$ solution (e.g., [MV03,Lyu08,LNSW13]) on public input $(\mathbf{A}_{(d)}, \mathbf{u})$, and prover's witness $\mathbf{z}$. However, constructing a ZKAoK of a message-signature pair $(d, \mathbf{z})$ is challenging, because on one hand, the prover has to convince the verifier that $\mathbf{A}_{(d)}\mathbf{z} = \mathbf{u} \bmod q$, while on the other hand, *both* $\mathbf{z}$ and d should be kept *secret* from the verifier.

Our first step towards solving the above challenge is to make the public verification matrix independent of d. Let $\overline{\mathbf{A}} = \left[\mathbf{A} | \mathbf{A}_0 | \mathbf{A}_1 | \ldots | \mathbf{A}_\ell\right] \in \mathbb{Z}_q^{n \times (\ell+2)m}$, and let $\mathbf{z} = (\mathbf{x} \| \mathbf{y})$, where $\mathbf{x}, \mathbf{y} \in \mathbb{Z}^m$, then we have:

$$\mathbf{u} = \mathbf{A}_{(d)}\mathbf{z} = \mathbf{A}\mathbf{x} + \mathbf{A}_0\mathbf{y} + \sum_{j=1}^{\ell} \mathbf{A}_i(d_i\mathbf{y}) = \overline{\mathbf{A}}\overline{\mathbf{z}} \bmod q,$$

where $\overline{\mathbf{z}} \in \mathbb{Z}^{(\ell+2)m}$ has the form $\overline{\mathbf{z}} = (\mathbf{x} \| \mathbf{y} \| d_1\mathbf{y} \| \ldots \| d_\ell\mathbf{y})$. Now our goal is: Given $(\overline{\mathbf{A}}, \mathbf{u})$, arguing in zero-knowledge the possession of $\overline{\mathbf{z}} \in \mathbb{Z}^{(\ell+2)m}$ such that:

[2] We note that such sequence of integers was previously used by Lipmaa et al. [LAN02] in the context of range proofs, but under a different representation: $\beta_j = \lfloor (\beta + 2^{j-1})/2^j \rfloor$ for each $j \in [p]$.

1. "$\|\overline{\mathbf{z}}\|_\infty \leq \beta$ and $\overline{\mathbf{A}}\overline{\mathbf{z}} = \mathbf{u} \bmod q$." This part can be done using the Decomposition-Extension technique from [LNSW13] for an ISIS solution. Specifically, we transform $\mathbf{x}$ and $\mathbf{y}$ into $p = \lfloor \log \beta \rfloor + 1$ vectors $\mathbf{x}_1, \ldots, \mathbf{x}_p \in \mathsf{B}_{3m}$ and $\mathbf{y}_1, \ldots, \mathbf{y}_p \in \mathsf{B}_{3m}$, respectively.
2. "$\overline{\mathbf{z}}$ has the form $\overline{\mathbf{z}} = (\mathbf{x}\|\mathbf{y}\|d_1\mathbf{y}\| \ldots \|d_\ell\mathbf{y})$ for certain $d \in \{0,1\}^\ell$." At a high level, to argue that $d \in \{0,1\}^\ell$, we first extend d to $d^* = (d_1, \ldots, d_\ell, d_{\ell+1}, \ldots, d_{2\ell}) \in \mathsf{B}_{2\ell}$, where $\mathsf{B}_{2\ell}$ is the set of all vectors in $\{0,1\}^{2\ell}$ having Hamming weight ℓ, and then show that a random permutation of d^* belongs to the set $\mathsf{B}_{2\ell}$, which implies that the original $d \in \{0,1\}^\ell$.

Now, for simplicity of description of our technique, we introduce the following notations:

- For permutations $\pi, \psi \in \mathsf{S}_{3m}$; $\tau \in \mathsf{S}_{2\ell}$, and for vector $\mathbf{t} = (\mathbf{t}_{-1}\|\mathbf{t}_0\|\mathbf{t}_1\| \ldots \|\mathbf{t}_{2\ell}) \in \mathbb{Z}_q^{(2\ell+2)3m}$ consisting of $(2\ell+2)$ blocks of size $3m$, we define:

$$F_{\pi,\psi,\tau}(\mathbf{t}) = \big(\pi(\mathbf{t}_{-1})\|\psi(\mathbf{t}_0)\|\psi(\mathbf{t}_{\tau(1)})\|\psi(\mathbf{t}_{\tau(2)})\| \ldots \|\psi(\mathbf{t}_{\tau(2\ell)})\big).$$

  Namely, $F_{\pi,\psi,\tau}(\mathbf{t})$ is a composition of 3 permutations. It *rearranges* the order of the 2ℓ blocks $\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{2\ell}$ according to τ, and then *permutes* block $\mathbf{t}_{-1}$ according to π, and the other $(2\ell+1)$ blocks according to ψ.
- Given $e = (e_1, e_2, \ldots, e_{2\ell}) \in \{0,1\}^{2\ell}$, we say that vector $\mathbf{t} \in \mathsf{VALID}(e)$ if $\mathbf{t} \in \{-1,0,1\}^{(2\ell+2)3m}$, and there exist $\mathbf{v}, \mathbf{w} \in \mathsf{B}_{3m}$ such that $\mathbf{t} = (\mathbf{v}\|\mathbf{w}\|e_1\mathbf{w}\|e_2\mathbf{w}\| \ldots \|e_{2\ell}\mathbf{w})$.

We now describe our technique. We define the sequence $\beta_1, \ldots, \beta_p$ as in [LNSW13], and let:

$$\mathbf{A}^* = \big[\, \mathbf{A} \,|\, 0^{n\times 2m} \,|\, \mathbf{A}_0 \,|\, 0^{n\times 2m} \,|\, \mathbf{A}_1 \,|\, 0^{n\times 2m} \,|\, \ldots \,|\, \mathbf{A}_\ell \,|\, 0^{n\times 2m} \,|\, 0^{n\times 3m\ell} \,\big] \in \mathbb{Z}_q^{n\times(2\ell+2)3m}, \quad (1)$$

$$\mathbf{z}_j = \big(\, \mathbf{x}_j \,\|\, \mathbf{y}_j \,\|\, d_1\mathbf{y}_j \,\|\, \ldots \,\|\, d_\ell\mathbf{y}_j \,\|\, d_{\ell+1}\mathbf{y}_j \,\|\, \ldots \,\|\, d_{2\ell}\mathbf{y}_j \,\big) \in \{-1,0,1\}^{(2\ell+2)3m}, \ \forall j \in [p]. \quad (2)$$

We then have: $\mathbf{A}^*(\sum_{j=1}^p \beta_j\mathbf{z}_j) = \mathbf{u} \bmod q$, and $\mathbf{z}_j \in \mathsf{VALID}(d^*)$ for all $j \in [p]$. In Stern's framework, we proceed as follows:

- To argue that $\mathbf{A}^*(\sum_{j=1}^p \beta_j\mathbf{z}_j) = \mathbf{u} \bmod q$, we instead show that

$$\mathbf{A}^* \sum_{j=1}^{p} \beta_j(\mathbf{z}_j + \mathbf{r}_{\mathbf{z}}^{(j)}) - \mathbf{u} = \mathbf{A}^*(\sum_{j=1}^{p} \beta_j\mathbf{r}_{\mathbf{z}}^{(j)}) \bmod q,$$

  where $\mathbf{r}_{\mathbf{z}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{z}}^{(p)} \in \mathbb{Z}_q^{n\times(2\ell+2)3m}$ are uniformly random "masking" vectors for the $\mathbf{z}_j$'s.
- We sample a uniformly random permutation $\tau \in \mathsf{S}_{2\ell}$, and for each $j \in [p]$, sample uniformly random $\pi_j, \psi_j \in \mathsf{S}_{3m}$, and send $\mathbf{t}_d = \tau(d^*)$ together with $\mathbf{t}_{\mathbf{z}}^{(j)} = F_{\pi_j,\psi_j,\tau}(\mathbf{z}_j)$, for all j. Seeing that $\mathbf{t}_d \in \mathsf{B}_{2\ell}$, and $\mathbf{t}_{\mathbf{z}}^{(j)} \in \mathsf{VALID}(\mathbf{t}_d)$, the verifier will be convinced that $\mathbf{z}_j \in \mathsf{VALID}(d^*)$ while learning no additional information about $\mathbf{z}_j$ or d^*.

Based on the above discussion, we can build a ZKAoK of a valid message-signature pair for Boyen's signature scheme. For convenience, we will present the details in the combined protocol in Section 3.3.

3.2 A Lattice-based Verifiable Encryption Protocol

We consider two lattice-based encryption schemes:

1. The GPV-IBE scheme [GPV08] based on LWE, to be employed in the group signature in Section 4.
2. The LPR encryption scheme [LPR13] based on $\mathsf{Ring\text{-}LWE}$, to be employed in the ring-based group signature in Section 5.

We observe that, in both of these schemes, if one encrypts a plaintext $d \in \{0,1\}^\ell$ using the HNF variants of LWE and Ring-LWE, respectively, then the relation among the related objects can be expressed as:

$$\mathbf{Pe} + (\,0^{k_1-\ell} \,\|\, \lfloor q/2 \rfloor d\,) = \mathbf{c} \bmod q,$$

where $\mathbf{P} \in \mathbb{Z}_q^{k_1 \times k_2}$ is a matrix obtained from the public key, $\mathbf{c} \in \mathbb{Z}_q^{k_1}$ is a ciphertext, $\mathbf{e} \in \mathbb{Z}^{k_2}$ is the encryption randomness satisfying $\|\mathbf{e}\|_\infty \leq b$. Here k_1, k_2, b are certain parameters depending on the underlying scheme.

Our goal is to construct a verifiable encryption protocol for both of the mentioned above schemes, namely, a protocol such that: given $(\mathbf{P}, \mathbf{c})$, the prover, possessing $(\mathbf{e}, d)$, can argue in zero-knowledge that $\mathbf{c}$ is a correct encryption of d. We observe that, this task can be achieved by adapting the Decomposition-Extension technique by Ling et al., as follows:

- To argue that $d \in \{0,1\}^\ell$, we can use the same technique as in the previous section, i.e., extend d to $d^* \in \mathsf{B}_{2\ell}$, then use a random permutation.
- To argue that $\mathbf{e} \in \mathbb{Z}^{k_2}$ and $\|\mathbf{e}\|_\infty \leq b$, we form the vectors $\mathbf{e}_1, \ldots, \mathbf{e}_{\bar{p}} \in \mathsf{B}_{3k_2}$, where $\bar{p} = \lfloor \log b \rfloor + 1$, then use random permutations to show the membership of the $\mathbf{e}_j$'s in B_{3k_2}.
- Next, we define the following two extended matrices:

$$\mathbf{P}^* = \left[\, \mathbf{P} \mid 0^{k_1 \times 2k_2} \,\right] \in \mathbb{Z}_q^{k_1 \times 3k_2}; \quad \mathbf{Q} = \begin{pmatrix} 0^{(k_1-\ell)\times \ell} & | & 0^{(k_1-\ell)\times \ell} \\ ------ & & ------ \\ \lfloor q/2 \rfloor \mathbf{I}_\ell & | & 0^{\ell \times \ell} \end{pmatrix} \in \{0, \lfloor q/2 \rfloor\}^{k_1 \times 2\ell}. \quad (3)$$

- We then have that:

$$\mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j \mathbf{e}_j\big) + \mathbf{Q}d^* = \mathbf{Pe} + (\,0^{k_1-\ell} \,\|\, \lfloor q/2 \rfloor d\,) = \mathbf{c} \bmod q. \quad (4)$$

In Stern's framework, to argue that (4) is true, we instead show that:

$$\mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j(\mathbf{e}_j + \mathbf{r}_\mathbf{e}{}^{(j)})\big) + \mathbf{Q}(d^* + \mathbf{r}_d) - \mathbf{c} = \mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j \mathbf{r}_\mathbf{e}^{(j)}\big) + \mathbf{Q}\mathbf{r}_d \bmod q,$$

where $\mathbf{r}_\mathbf{e}^{(j)} \in \mathbb{Z}_q^{3k_2}$, for every $j \in [\bar{p}]$, and $\mathbf{r}_d \in \mathbb{Z}_q^{2\ell}$ are uniformly random masking vectors.

3.3 The Combined Protocol

We now describe in detail the combined protocol that allows the prover to argue that it knows a valid message-signature pair $(d, \mathbf{z})$ for Boyen's signature scheme, and that a given ciphertext correctly encrypts d. The associated relation $\mathrm{R}_{\mathsf{gs}}(n, \ell, q, m, k_1, k_2, \beta, b)$ is defined as follows.

Definition 6.

$$\mathrm{R}_{\mathsf{gs}} = \Big\{ \Big(\big(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n\times m}; \mathbf{u} \in \mathbb{Z}_q^n; \mathbf{P} \in \mathbb{Z}_q^{k_1 \times k_2}; \mathbf{c} \in \mathbb{Z}_q^{k_1}\big); d \in \{0,1\}^\ell; \mathbf{z} \in \mathbb{Z}^{2m}; \mathbf{e} \in \mathbb{Z}^{k_2} \Big) :$$

$$\big(\|\mathbf{z}\|_\infty \leq \beta \wedge \Big[\, \mathbf{A} \,\Big|\, \mathbf{A}_0 + \sum_{i=1}^{\ell} d_i \mathbf{A}_i \,\Big] \mathbf{z} = \mathbf{u} \bmod q\big) \wedge \big(\|\mathbf{e}\|_\infty \leq b \wedge \mathbf{Pe} + (\,0^{k_1-\ell} \,\|\, \lfloor q/2 \rfloor d\,) = \mathbf{c} \bmod q\big). \Big\}$$

Let COM be the statistically hiding and computationally binding string commitment scheme from [KTX08]. Let $p = \lfloor \log \beta \rfloor + 1$ and $\bar{p} = \lfloor \log b \rfloor + 1$ and define two sequences of integers $\beta_1, \ldots, \beta_p$ and $b_1, \ldots, b_{\bar{p}}$ as in sections [LNSW13]. The inputs of two parties are as follows:

– The common input is $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c})$. Both parties form matrices $\mathbf{A}^*$, $\mathbf{P}^*$, $\mathbf{Q}$ as described in (1) and (3).
– The prover's witness is $(d, \mathbf{z}, \mathbf{e})$. Using the techniques above, the prover extends d to some $d^* \in \mathsf{B}_{2\ell}$ and forms vectors $\mathbf{z}_1, \ldots, \mathbf{z}_p \in \mathsf{VALID}(d^*)$, and $\mathbf{e}_1, \ldots, \mathbf{e}_{\bar{p}} \in \mathsf{B}_{3k_2}$. The obtained vectors satisfy:

$$\mathbf{A}^*\big(\sum_{j=1}^{p} \beta_j \mathbf{z}_j\big) = \mathbf{u} \bmod q \;\;\wedge\;\; \mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j \mathbf{e}_j\big) + \mathbf{Q}d^* = \mathbf{c} \bmod q.$$

The interaction between P and V is described in Figure 1.

The following theorem summarizes the properties of the above protocol.

Theorem 1. *Let* COM *be a statistically hiding and computationally binding string commitment scheme. Then the protocol in Figure 1 is a statistical zero-knowledge argument of knowledge for the relation* $\mathrm{R}_{\mathsf{gs}}(n, \ell, q, m, k_1, k_2, \beta, b)$*. Each round of the protocol has perfect completeness, soundness error* $2/3$*, and communication cost* $(\mathcal{O}(\ell m)\log\beta + \mathcal{O}(k_2)\log b)\log q$*.*

The proof of Theorem 1 employs the standard proof technique for Stern-type protocols. It is given in Appendix B.

4 An Improved Lattice-based Group Signature Scheme

4.1 Description of Our Scheme

We first specify the parameters of the scheme. Let n be the security parameter, and let $N = 2^\ell = \mathsf{poly}(n)$ be the maximum expected number of group users. Then we choose other scheme parameters such that Boyen's signature scheme and the GPV-IBE scheme function properly, and are secure. Specifically, let modulus $q = \mathcal{O}(\ell \cdot n^2)$ be prime, dimension $m \geq 2n\log q$, and Gaussian parameter $s = \omega(\log m)$. The infinity norm bound for signatures from Boyen's scheme is integer $\beta = \widetilde{\mathcal{O}}(\sqrt{\ell n})$. The norm bound for LWE noises is integer b such that $q/b = \ell\widetilde{\mathcal{O}}(n)$.

Choose hash functions $\mathcal{H}_1 : \{0,1\}^* \to \mathbb{Z}_q^{n\times\ell}$ and $\mathcal{H}_2 : \{0,1\}^* \to \{1,2,3\}^t$, to be modeled as random oracles, and select a one-time signature scheme $\mathcal{OTS} = (\mathsf{OGen}, \mathsf{OSign}, \mathsf{OVer})$. Let χ be a b-bounded distribution over $\mathbb{Z}$.

Our group signature scheme is described as follows:

KeyGen$(1^n, 1^N)$**:** This algorithm performs the following steps:

1. Generate verification key $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$ and signing key $\mathbf{T_A}$ for Boyen's signature scheme (see Section 2.3 for more details). Then for each $d = (d_1, \ldots, d_\ell) \in \{0,1\}^\ell$, use $\mathbf{T_A}$ to generate $\mathsf{gsk}[d]$ as a Boyen signature on message d.
2. Generate encrypting and decrypting keys for the GPV-IBE scheme: Run algorithm $\mathsf{GenTrap}(n, m, q)$ from [GPV08] to output $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$ together with a trapdoor basis $\mathbf{T_B}$ for $\Lambda^\perp(\mathbf{B})$.
3. Output

$$\mathsf{gpk} = \big((\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}),\ \mathbf{B}\big);\ \ \mathsf{gmsk} = \mathbf{T_B};\ \ \mathsf{gsk} = \{\mathsf{gsk}[d]\}_{d\in\{0,1\}^\ell}.$$

Sign$(\mathsf{gsk}[d], M)$: Given gpk, to sign a message $M \in \{0,1\}^*$ using the secret key $\mathsf{gsk}[d] = \mathbf{z}$, the user generates a key pair $(\mathsf{ovk}, \mathsf{osk}) \leftarrow \mathsf{OGen}(1^n)$ for $\mathcal{OTS}$, and then performs the following steps:

1. Encrypt the index d with respect to "identity" ovk as follows. Let $\mathbf{G} = \mathcal{H}_1(\mathsf{ovk}) \in \mathbb{Z}_q^{n\times\ell}$. Sample $\mathbf{s} \hookleftarrow \chi^n; \mathbf{e}_1 \hookleftarrow \chi^m; \mathbf{e}_2 \hookleftarrow \chi^\ell$, then compute the ciphertext:

$$\big(\mathbf{c}_1 = \mathbf{B}^T\mathbf{s} + \mathbf{e}_1, \mathbf{c}_2 = \mathbf{G}^T\mathbf{s} + \mathbf{e}_2 + \lfloor q/2 \rfloor d\big) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^\ell.$$

1. **Commitment:** P samples

$$\begin{cases} \mathbf{r}_{\mathbf{z}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{z}}^{(p)} \xleftarrow{\$} \mathbb{Z}_q^{(2\ell+2)3m};\ \mathbf{r}_{\mathbf{e}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{e}}^{(\bar{p})} \xleftarrow{\$} \mathbb{Z}_q^{3k_2};\ \mathbf{r}_d \xleftarrow{\$} \mathbb{Z}_q^{2\ell} \\ \tau \xleftarrow{\$} \mathsf{S}_{2\ell};\ \pi_1, \ldots, \pi_p, \psi_1, \ldots, \psi_p \xleftarrow{\$} \mathsf{S}_{3m};\ \phi_1, \ldots, \phi_{\bar{p}} \xleftarrow{\$} \mathsf{S}_{3k_2}. \end{cases}$$

   Then P sends the commitment $\mathrm{CMT} = \big(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3\big)$ to V, where

$$\begin{cases} \mathbf{c}_1 = \mathsf{COM}\big(\tau;\ \{\pi_j\}_{j=1}^{p};\ \{\psi_j\}_{j=1}^{p};\ \{\phi_j\}_{j=1}^{\bar{p}};\ \mathbf{A}^*\big(\sum_{j=1}^{p} \beta_j \mathbf{r}_{\mathbf{z}}^{(j)}\big);\ \mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j \mathbf{r}_{\mathbf{e}}^{(j)}\big) + \mathbf{Q}\mathbf{r}_d\big), \\ \mathbf{c}_2 = \mathsf{COM}\big(\{F_{\pi_j,\psi_j,\tau}(\mathbf{r}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\ \{\phi_j(\mathbf{r}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\ \tau(\mathbf{r}_d)\big), \\ \mathbf{c}_3 = \mathsf{COM}\big(\{F_{\pi_j,\psi_j,\tau}(\mathbf{z}_j + \mathbf{r}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\ \{\phi_j(\mathbf{e}_j + \mathbf{r}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\ \tau(d^* + \mathbf{r}_d)\big). \end{cases} \tag{5}$$

2. **Challenge:** V sends a challenge $Ch \xleftarrow{\$} \{1, 2, 3\}$ to P.
3. **Response:** Depending on Ch, P computes the response RSP as follows:
   - Case $Ch = 1$: For each $j \in [p]$, let $\mathbf{t}_{\mathbf{z}}^{(j)} = F_{\pi_j,\psi_j,\tau}(\mathbf{z}_j)$ and $\mathbf{v}_{\mathbf{z}}^{(j)} = F_{\pi_j,\psi_j,\tau}(\mathbf{r}_{\mathbf{z}}^{(j)})$. For each $j \in [\bar{p}]$, let $\mathbf{t}_{\mathbf{e}}^{(j)} = \phi_j(\mathbf{e}_j)$ and $\mathbf{v}_{\mathbf{e}}^{(j)} = \phi_j(\mathbf{r}_{\mathbf{e}}^{(j)})$. Let $\mathbf{t}_d = \tau(d^*)$ and $\mathbf{v}_d = \tau(\mathbf{r}_d)$. Then the prover sends:

$$\mathrm{RSP} = \big(\{\mathbf{t}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{v}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{t}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \{\mathbf{v}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{t}_d;\ \mathbf{v}_d\big). \tag{6}$$

   - Case $Ch = 2$: For each $j \in [p]$, let $\widehat{\pi}_j = \pi_j$; $\widehat{\psi}_j = \psi_j$; and $\mathbf{w}_{\mathbf{z}}^{(j)} = \mathbf{z}_j + \mathbf{r}_{\mathbf{z}}^{(j)}$. For each $j \in [\bar{p}]$, let $\widehat{\phi}_j = \phi_j$; and $\mathbf{w}_{\mathbf{e}}^{(j)} = \mathbf{e}_j + \mathbf{r}_{\mathbf{e}}^{(j)}$. Let $\widehat{\tau} = \tau$ and $\mathbf{w}_d = d^* + \mathbf{r}_d$. Then the prover sends:

$$\mathrm{RSP} = \big(\widehat{\tau};\ \{\widehat{\pi}_j\}_{j=1}^{p};\ \{\widehat{\psi}_j\}_{j=1}^{p};\ \{\widehat{\phi}_j\}_{j=1}^{\bar{p}};\ \{\mathbf{w}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{w}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{w}_d\big). \tag{7}$$

   - Case $Ch = 3$: For each $j \in [p]$, let $\widetilde{\pi}_j = \pi_j$; $\widetilde{\psi}_j = \psi_j$; and $\mathbf{y}_{\mathbf{z}}^{(j)} = \mathbf{r}_{\mathbf{z}}^{(j)}$. For each $j \in [\bar{p}]$, let $\widetilde{\phi}_j = \phi_j$; and $\mathbf{y}_{\mathbf{e}}^{(j)} = \mathbf{r}_{\mathbf{e}}^{(j)}$. Let $\widetilde{\tau} = \tau$ and $\mathbf{y}_d = \mathbf{r}_d$. Then the prover sends:

$$\mathrm{RSP} = \big(\widetilde{\tau};\ \{\widetilde{\pi}_j\}_{j=1}^{p};\ \{\widetilde{\psi}_j\}_{j=1}^{p};\ \{\widetilde{\phi}_j\}_{j=1}^{\bar{p}};\ \{\mathbf{y}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{y}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{y}_d\big). \tag{8}$$

Verification: Receiving RSP, the verifier proceeds as follows:

- Case $Ch = 1$: Parse RSP as in (6). Check that $\mathbf{t}_d \in \mathsf{B}_{2\ell}$; $\mathbf{t}_{\mathbf{z}}^{(j)} \in \mathsf{VALID}(\mathbf{t}_d)$, $\forall j \in [p]$; $\mathbf{t}_{\mathbf{e}}^{(j)} \in \mathsf{B}_{3k_2}$, $\forall j \in [\bar{p}]$; and that

$$\begin{cases} \mathbf{c}_2 = \mathsf{COM}\big(\{\mathbf{v}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{v}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{v}_d\big) \\ \mathbf{c}_3 = \mathsf{COM}\big(\{\mathbf{t}_{\mathbf{z}}^{(j)} + \mathbf{v}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{t}_{\mathbf{e}}^{(j)} + \mathbf{v}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{t}_d + \mathbf{v}_d\big). \end{cases}$$

- Case $Ch = 2$: Parse RSP as in (7). Check that:

$$\begin{cases} \mathbf{c}_1 = \mathsf{COM}\big(\widehat{\tau};\ \{\widehat{\pi}_j\}_{j=1}^{p};\ \{\widehat{\psi}_j\}_{j=1}^{p};\ \{\widehat{\phi}_j\}_{j=1}^{\bar{p}};\ \mathbf{A}^*\big(\sum_{j=1}^{p} \beta_j \mathbf{w}_{\mathbf{z}}^{(j)}\big) - \mathbf{u};\ \ \mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_i \mathbf{w}_{\mathbf{e}}^{(j)}\big) + \mathbf{Q}\mathbf{w}_d - \mathbf{c}\big), \\ \mathbf{c}_3 = \mathsf{COM}\big(\ \{F_{\widehat{\pi}_j,\widehat{\psi}_j,\widehat{\tau}}(\mathbf{w}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\ \{\widehat{\phi}_j(\mathbf{w}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\ \widehat{\tau}(\mathbf{w}_d)\big). \end{cases}$$

- Case $Ch = 3$: Parse RSP as in (8). Check that:

$$\begin{cases} \mathbf{c}_1 = \mathsf{COM}\big(\widetilde{\tau};\ \{\widetilde{\pi}_j\}_{j=1}^{p};\ \{\widetilde{\psi}_j\}_{j=1}^{p};\ \{\widetilde{\phi}_j\}_{j=1}^{\bar{p}};\ \mathbf{A}^*\big(\sum_{j=1}^{p} \beta_j \mathbf{y}_{\mathbf{z}}^{(j)}\big);\ \ \mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_i \mathbf{y}_{\mathbf{e}}^{(j)}\big) + \mathbf{Q}\mathbf{y}_d\big), \\ \mathbf{c}_2 = \mathsf{COM}\big(\ \{F_{\widetilde{\pi}_j,\widetilde{\psi}_j,\widetilde{\tau}}(\mathbf{y}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\ \{\widetilde{\phi}_j(\mathbf{y}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\ \widetilde{\tau}(\mathbf{y}_d)\big). \end{cases}$$

In each case, V outputs 1 if and only if all the conditions hold. Otherwise, it outputs 0.

Fig. 1: A zero-knowledge argument that the prover possesses a valid message-signature pair $(d, \mathbf{z})$ for Boyen's signature scheme, and that a given ciphertext correctly encrypts d.

2. Generate a $\mathsf{NIZKAoK}$ Π to show the possession of a valid message-signature pair $(d, \mathbf{z})$ for Boyen's signature, and that $(\mathbf{c}_1, \mathbf{c}_2)$ is a correct GPV-IBE encryption of d with respect to "identity" ovk. This is done as follows:

– Let $k_1 := m + \ell$ and $k_2 := n + m + \ell$, and form the following:

$$\mathbf{P} = \left(\begin{array}{c|c} \mathbf{B}^T & \\ \hline \mathbf{G}^T & \mathbf{I}_{m+\ell} \end{array}\right) \in \mathbb{Z}_q^{k_1 \times k_2}; \quad \mathbf{c} = \begin{pmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{pmatrix} \in \mathbb{Z}^{k_1}; \quad \mathbf{e} = \begin{pmatrix} \mathbf{s} \\ \mathbf{e}_1 \\ \mathbf{e}_2 \end{pmatrix} \in \mathbb{Z}^{k_2}, \quad (9)$$

Then we have $\|\mathbf{e}\|_\infty \leq b$, and $\mathbf{Pe} + (\,0^{k_1-\ell} \,\|\, \lfloor q/2 \rfloor d\,) = \mathbf{c} \bmod q$. Now one can observe that:

$$\big((\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c}), d, \mathbf{z}, \mathbf{e}\big) \in \mathrm{R}_{\mathsf{gs}}(n, \ell, q, m, k_1, k_2, \beta, b).$$

– Run the protocol in Section 3.3 with public input $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c})$ and prover's witness $(d, \mathbf{z}, \mathbf{e})$. The protocol is repeated $t = \omega(\log n)$ times to make the soundness error negligibly small, and then made non-interactive using the Fiat-Shamir heuristic as a triple $\Pi = \big(\{\mathrm{CMT}_j\}_{j=1}^t, \mathrm{CH}, \{\mathrm{RSP}_j\}_{j=1}^t\big)$, where $\mathrm{CH} = \{Ch_j\}_{j=1}^t = \mathcal{H}_2\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$.

3. Compute a one-time signature $sig = \mathsf{OSign}(\mathsf{osk}; \mathbf{c}_1, \mathbf{c}_2, \Pi)$.
4. Output the group signature $\Sigma = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi, sig\big)$.

Verify$(\mathsf{gpk}, M, \Sigma)$: This algorithm works as follows:

1. Parse Σ as $\big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi, sig\big)$. If $\mathsf{OVer}(\mathsf{ovk}; sig; (\mathbf{c}_1, \mathbf{c}_2), \Pi) = 0$ then return 0.
2. Parse Π as $\big(\{\mathrm{CMT}_j\}_{j=1}^t, \{Ch_j\}_{j=1}^t, \{\mathrm{RSP}_j\}_{j=1}^t\big)$.
   If $\big(Ch_1, \ldots, Ch_t\big) \neq \mathcal{H}_2\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$, then return 0.
3. Compute $\mathbf{G} = \mathcal{H}_1(\mathsf{ovk})$ and form $\mathbf{P}, \mathbf{c}$ as in (9). Then for $j = 1$ to t, run the verification step of the protocol from Section 3.3 with public input $\big(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c}\big)$ to check the validity of RSP_j with respect to CMT_j and Ch_j. If any of the conditions does not hold, then return 0.
4. Return 1.

Open$(\mathsf{gmsk}, M, \Sigma)$ On input $\mathsf{gmsk} = \mathbf{T_B}$ and a signature $\Sigma = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi, sig\big)$, this algorithm decrypts $(\mathbf{c}_1, \mathbf{c}_2)$ as follows:

1. Extract the decryption key for "identity" ovk: Let $\mathbf{G} = [\mathbf{g}_1 | \ldots | \mathbf{g}_\ell] = \mathcal{H}_1(\mathsf{ovk})$. Then for $i \in [\ell]$, sample $\mathbf{y}_i \hookleftarrow \mathsf{SamplePre}(\mathbf{T_B}, \mathbf{B}, \mathbf{g}_i, s)$ (see [GPV08]), and let $\mathbf{Y} = [\mathbf{y}_1 | \ldots | \mathbf{y}_\ell] \in \mathbb{Z}^{m \times \ell}$.
2. Compute $d' = (d'_1, \ldots, d'_\ell) = \mathbf{c}_2 - \mathbf{Y}^T \mathbf{c}_1 \in \mathbb{Z}_q^\ell$. For each $i \in [\ell]$, if d'_i is closer to 0 than to $\lfloor q/2 \rfloor$ modulo q, then let $d_i = 0$; otherwise, let $d_i = 1$.
3. Return $d = (d_1, \ldots, d_\ell) \in \{0,1\}^\ell$.

4.2 Analysis of the Scheme

Efficiency and Correctness. The given group signature scheme can be implemented in polynomial time. The bit-size of the $\mathsf{NIZKAoK}$ Π is roughly $t = \omega(\log n)$ times the communication cost of the interactive protocol in Section 3.3, which is $\widetilde{\mathcal{O}}(\ell n)$ for the chosen parameters. This is also the asymptotical bound on the size of the group signature Σ.

The correctness of algorithm Verify follows from the facts that every group user with a valid secret key is able to compute a satisfying witness for the relation $\mathrm{R}_{\mathsf{gs}}(n, \ell, q, m, k_1, k_2, \beta, b)\big)$, and that the underlying argument system is perfectly complete. Moreover, we set the parameters so that the GPV-IBE scheme is correct, which implies that algorithm Open is also correct.

Theorem 2 (CCA-anonymity). *Suppose that $\mathcal{OTS}$ is a strongly unforgeable one-time signature. In the random oracle model, the group signature scheme described in Section 4.1 is CCA-anonymous if the $\mathsf{LWE}_{n,q,\chi}$ problem is hard.*

As a corollary, the CCA-anonymity of the scheme can be based on the quantum worst-case hardness of SIVP_γ, with $\gamma = \widetilde{\mathcal{O}}(nq/b) = \ell\widetilde{\mathcal{O}}(n^2)$.

Proof. Let $\mathcal{A}$ be any PPT adversary attacking the CCA-anonymity of the scheme with advantage ϵ. Using the strong unforgeability of $\mathcal{OTS}$, the statistical ZK property of the underlying argument system, and the LWE assumption, we will prove that $\epsilon = \mathsf{negl}(n)$. Specifically, we construct a sequence of indistinguishable experiments $G_0^{(b)}, G_1^{(b)}, G_2^{(b)}, G_3^{(b)}, G_4^{(b)}, G_5$, such that, $\mathbf{Adv}_{\mathcal{A}}(G_0^{(b)}) = \epsilon$ and $\mathbf{Adv}_{\mathcal{A}}(G_5) = 0$.

Experiment $G_0^{(b)}$. This is the real CCA-anonymity game. The challenger runs $\mathsf{KeyGen}(1^n, 1^N)$ to obtain $(\mathsf{gpk}, \mathsf{gmsk} = \mathbf{T_B}, \{\mathsf{gsk}[d]\}_{d\in\{0,1\}^\ell})$, and then gives gpk and $\{\mathsf{gsk}[d]\}_{d\in\{0,1\}^\ell}$ to $\mathcal{A}$. Using the decryption key $\mathbf{T_B}$, the challenger can answer all the signature opening queries. In the challenge phase, $\mathcal{A}$ sends a message M together with two indices $d_0, d_1 \in \{0,1\}^\ell$. The challenger sends back a challenge signature $\Sigma^* = \big(\mathsf{ovk}^*, (\mathbf{c}_1^*, \mathbf{c}_2^*), \Pi^*, sig^*\big) \leftarrow \mathsf{Sign}(\mathsf{gpk}, \mathsf{gsk}[d_b])$. The adversary then outputs $b' \in \{0,1\}$. The experiment returns 1 if $b' = b$ or 0 otherwise. We remark that, in this experiment, all the queries to random oracles $\mathcal{H}_1$ and $\mathcal{H}_2$ are responded with truly uniformly random elements in the respective ranges. By assumption, $\mathcal{A}$ has advantage ϵ in this experiment.

Experiment $G_1^{(b)}$. In this experiment, we make a slight modification with respect to $G_0^{(b)}$: the one-time signature key pair $(\mathsf{ovk}^*, \mathsf{osk}^*)$ is generated in the start of the experiment. During the game, if $\mathcal{A}$ requests for opening of valid signatures of the form $\Sigma = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi, sig\big)$, where $\mathsf{ovk} = \mathsf{ovk}^*$ then the challenger outputs a random bit and aborts. We will demonstrate that the strong unforgeability of $\mathcal{OTS}$ implies that experiments $G_1^{(b)}$ and $G_0^{(b)}$ are indistinguishable. Indeed, before the challenge phase, ovk^* is independent of $\mathcal{A}$'s view, and thus, the probability that ovk^* shows up in $\mathcal{A}$'s requests is negligible. On the other hand, after seeing the challenge signature $\Sigma^* = \big(\mathsf{ovk}^*, (\mathbf{c}_1^*, \mathbf{c}_2^*), \Pi^*, sig^*\big)$, if $\mathcal{A}$ comes up with a valid signature $\Sigma = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi, sig\big)$ such that $\mathsf{ovk} = \mathsf{ovk}^*$, then sig is a forged one-time signature, which violates the strong unforgeability of $\mathcal{OTS}$. Therefore, the probability that the challenger aborts in this experiment is negligible. Without loss of generality, in the subsequent experiments, we assume that $\mathcal{A}$ does not request for opening of valid signatures that include ovk^*.

Experiment $G_2^{(b)}$. In this experiment, we modify the generation of the encrypting matrices $\mathbf{B}$ and $\mathbf{G}$ and program the random oracle $\mathcal{H}_1$ accordingly. Instead of generating $\mathbf{B}$ with a trapdoor, and then computing $\mathbf{G}$ based on the trapdoor, we use uniformly random $\mathbf{B}^* \in \mathbb{Z}_q^{n\times m}$, and $\mathbf{G}^* \in \mathbb{Z}_q^{n\times\ell}$. The distribution of $(\mathbf{B}^*, \mathbf{G}^*)$ is statistically close to what in the real attack game (see, e.g., [GPV08]). In the challenge phase, the challenger programs $\mathcal{H}_1(\mathsf{ovk}^*) = \mathbf{G}^*$, computes ciphertext $(\mathbf{c}_1^*, \mathbf{c}_2^*)$, and generates the challenge signature Σ^* as in the previous experiments. To answer requests for opening of signature $\Sigma = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi, sig\big)$, the challenger samples a "decrypting matrix" $\mathbf{Y} \hookleftarrow (D_{\mathbb{Z}^m,\sigma_0})^\ell$, computes $\mathbf{G} = \mathbf{B}^*\mathbf{Y} \in \mathbb{Z}_q^{n\times\ell}$, programs $\mathcal{H}_1(\mathsf{ovk}) = \mathbf{G}$, and uses $\mathbf{G}$ for opening Σ. The challenger also locally records $(\mathsf{ovk}, \mathbf{Y}, \mathbf{G})$ to be reused in case $\mathcal{A}$ repeats the request for $\mathcal{H}_1(\mathsf{ovk})$. The distribution of $\mathbf{G}$ is statistically close to uniform over $\mathbb{Z}_q^{n\times\ell}$ (see, e.g., [GPV08]). It then follows that this experiment is indistinguishable from $G_1^{(b)}$.

Experiment $G_3^{(b)}$. In this experiment, instead of faithfully generating the NIZKAoK Π^*, the challenger simulates it without using the witness. This is done by running the simulator for the underlying interactive protocol for each $j \in [t]$, and then programming the random oracle $\mathcal{H}_2$ accordingly. The challenge signature $\Sigma^* = \big(\mathsf{ovk}^*, (\mathbf{c}_1^*, \mathbf{c}_2^*), \Pi^*, sig^*\big)$ is statistically close to the one in the previous experiments, because the argument system is statistically zero-knowledge. As a result, experiments $G_2^{(b)}$ and $G_3^{(b)}$ are indistinguishable.

Experiment $G_4^{(b)}$. In this experiment, we modify the generation of the ciphertext $(\mathbf{c}_1^*, \mathbf{c}_2^*)$. Recall that in experiment $G_3^{(b)}$, one has

$$(\mathbf{c}_1^* = (\mathbf{B}^*)^T\mathbf{s} + \mathbf{e}_1;\ \ \mathbf{c}_2^* = (\mathbf{G}^*)^T\mathbf{s} + \mathbf{e}_2 + \lfloor q/2 \rfloor d_b) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^\ell,$$

where $\mathbf{B}^* \in \mathbb{Z}_q^{n\times m}, \mathbf{G}^* \in \mathbb{Z}_q^{n\times \ell}, \mathbf{s} \in \mathbb{Z}_q^n$ are uniformly random, and $\mathbf{e}_1 \in \chi^m, \mathbf{e}_2 \in \chi^\ell$. Now we instead let $(\mathbf{c}_1^* = \mathbf{z}_1; \mathbf{c}_2^* = \mathbf{z}_2 + \lfloor q/2 \rfloor d_b)$, where $\mathbf{z}_1 \in \mathbb{Z}^m$ and $\mathbf{z}_2 \in \mathbb{Z}^\ell$ are uniformly random. The assumed hardness of the $\mathsf{LWE}_{n,q,\chi}$ problem (for the HNF variant [ACPS09]) implies that $G_3^{(b)}$ and $G_4^{(b)}$ are computationally indistinguishable. Indeed, if $\mathcal{A}$ can distinguish these two experiments, then it can also distinguish $\big(\mathbf{B}^*, (\mathbf{B}^*)^T\mathbf{s}+\mathbf{e}_1\big)$ and $\big(\mathbf{G}^*, (\mathbf{G}^*)^T\mathbf{s}+\mathbf{e}_2\big)$ from $(\mathbf{B}^*, \mathbf{z}_1)$ and $(\mathbf{G}^*, \mathbf{z}_2)$, respectively, which violates $\mathsf{LWE}_{n,q,\chi}$ assumption.

Experiment G_5. In this experiment we make a conceptual modification to $G_4^{(b)}$. Namely, we sample uniformly random $\mathbf{z}_1^{'} \in \mathbb{Z}_q^m$, $\mathbf{z}_2^{'} \in \mathbb{Z}_q^\ell$ and assign $\mathbf{c}_1^* = \mathbf{z}_1^{'}$, and $\mathbf{c}_2^* = \mathbf{z}_2^{'}$. It is clear that G_5 and $G_4^{(b)}$ are statistically indistinguishable. Moreover, since G_5 is no longer dependent on the challenger's bit b, the advantage of $\mathcal{A}$ in this experiment is 0.

It follows from the above construction that the advantage ϵ of $\mathcal{A}$ in attacking the CCA-anonymity of the scheme is negligible. This concludes the proof.

Theorem 3 (Traceability). *In the random oracle model, the group signature scheme described in Section 4.1 is fully traceable if the* $\mathsf{SIVP}_{\ell\cdot\widetilde{\mathcal{O}}(n^2)}$ *problem is hard.*

Proof. Without loss of generality, we assume that the string commitment scheme COM used in the underlying $\mathsf{NIZKAoK}$ is computationally binding, because an adversary breaking its computational binding property can be used to solve $\mathsf{SIVP}_{\ell\cdot\widetilde{\mathcal{O}}(n^2)}$.

Let $\mathcal{A}$ be an PPT traceability adversary against our group signature scheme with advantage ϵ, we construct a PPT forger $\mathcal{F}$ for Boyen's signature scheme whose advantage is polynomially related to ϵ. Since the unforgeability of Boyen's signature scheme can be based on the hardness of $\mathsf{SIVP}_{\ell\cdot\widetilde{\mathcal{O}}(n^2)}$ [Boy10,MP12], this completes the proof.

The forger $\mathcal{F}$ is given the verification key $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$ for Boyen's signature scheme. It then generates a key-pair $(\mathbf{B}, \mathbf{T_B})$ for the GPV IBE encryption scheme, and begins interacting with the adversary $\mathcal{A}$ by sending $\mathsf{gpk} = (\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{B})$ and $\mathsf{gsk} = \mathbf{T_B}$, the distribution of which is statistically close to that in the real attack game. Then $\mathcal{F}$ sets $CU = \emptyset$ and handles the queries from $\mathcal{A}$ as follows:

- Queries to the random oracles $\mathcal{H}_1$ and $\mathcal{H}_2$ are handled by consistently returning uniformly random values in the respective ranges. Suppose that $\mathcal{A}$ makes $Q_{\mathcal{H}_2}$ queries to $\mathcal{H}_2$, then for each $\kappa \leq Q_{\mathcal{H}_2}$, we let r_κ denote the answer to the κ-th query.
- Queries for the secret key $\mathsf{gsk}[d]$, for any $d \in \{0,1\}^\ell$: $\mathcal{F}$ queries its own signing oracle for Boyen's signature of d, and receives in return $\mathbf{z}_{(d)} \in \mathbb{Z}^{2m}$ such that $\|\mathbf{z}_{(d)}\|_\infty \leq \beta$ and $\mathbf{A}_{(d)}\mathbf{z}_{(d)} = \mathbf{u} \bmod q$, where $\mathbf{A}_{(d)}$ is computed in the usual way. Then $\mathcal{F}$ sets $CU := CU \cup \{d\}$ and sends $\mathbf{z}_{(d)}$ to $\mathcal{A}$.
- Queries for group signatures of user d on arbitrary message M: $\mathcal{F}$ returns with a simulated signature $\Sigma = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), \Pi', sig\big)$, where $(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), sig)$ are faithfully generated, while the $\mathsf{NIZKAoK}$ Π' is simulated without using the legitimate secret key (as in experiment $G_3^{(b)}$ in the proof of CCA anonymity). The zero-knowledge property of the underlying argument system guarantees that Σ is indistinguishable from a legitimate group signature.

Eventually $\mathcal{A}$ outputs a message M^* and a forged group signature

$$\Sigma^* = \big(\mathsf{ovk}, (\mathbf{c}_1, \mathbf{c}_2), (\{\mathrm{CMT}_j\}_{j=1}^t, \{Ch_j\}_{j=1}^t, \{\mathrm{RSP}_j\}_{j=1}^t), sig\big),$$

which satisfies the requirements of the traceability game. Then $\mathcal{F}$ exploits the forgery as follows. First, one can argue that $\mathcal{A}$ must have queried $\mathcal{H}_2$ on input $\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$, since otherwise, the probability that $\big(Ch_1, \ldots, Ch_t\big) = \mathcal{H}_2\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$ is at most 3^{-t}. Therefore, with probability at least $\epsilon - 3^{-t}$, there exists certain $\kappa^* \leq Q_{\mathcal{H}_2}$ such that the κ^*-th oracle query involves the tuple $\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$. Next, $\mathcal{F}$ picks κ^* as the target forking point and replays $\mathcal{A}$ many times with the same random tape and input as in the original run. In each rerun, for the first $\kappa^* - 1$ queries, $\mathcal{A}$ is given the same answers $r_1, \ldots, r_{\kappa^*-1}$ as in the initial run, but from the κ^*-th query onwards, $\mathcal{F}$ replies with fresh random values $r'_{\kappa^*}, \ldots, r'_{q_{\mathcal{H}_2}} \xleftarrow{\$} \{1,2,3\}^t$. The Improved Forking Lemma of Pointcheval and Vaudenay [PV97, Lemma 7] implies that, with probability larger than $1/2$, algorithm $\mathcal{F}$ can obtain a 3-fork involving the tuple $\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$ after less than $32 \cdot Q_{\mathcal{H}_2}/(\epsilon - 3^{-t})$ executions of $\mathcal{A}$. Now, let the answers of $\mathcal{F}$ with respect to the 3-fork branches be

$$r_{\kappa^*}^{(1)} = (Ch_1^{(1)}, \ldots, Ch_t^{(1)});\ r_{\kappa^*}^{(2)} = (Ch_1^{(2)}, \ldots, Ch_t^{(2)});\ r_{\kappa^*}^{(3)} = (Ch_1^{(3)}, \ldots, Ch_t^{(3)}).$$

A simple calculation shows that: $\Pr\big[\exists j \in \{1, \ldots, t\} : \{Ch_j^{(1)}, Ch_j^{(2)}, Ch_j^{(3)}\} = \{1,2,3\}\big] = 1 - (7/9)^t$. Conditioned on the existence of such j, one parses the 3 forgeries corresponding to the fork branches to obtain $\big(\mathrm{RSP}_j^{(1)}, \mathrm{RSP}_j^{(2)}, \mathrm{RSP}_j^{(3)}\big)$. They turn out to be 3 *valid* responses with respect to 3 different challenges for the same commitment CMT_j. Since COM is assumed to be computationally-binding, we can use the knowledge extractor of the underlying argument system to extract $(d^*, \mathbf{z}^*, \mathbf{s}^*, \mathbf{e}_1^*, \mathbf{e}_2^*) \in \{0,1\}^\ell \times \mathbb{Z}^{2m} \times \mathbb{Z}_q^n \times \mathbb{Z}^m \times \mathbb{Z}^\ell$ such that $\|\mathbf{z}^*\|_\infty \leq \beta$ and $\mathbf{A}_{(d^*)}\mathbf{z}^* = \mathbf{u} \bmod q$; and $\mathbf{s}^*, \mathbf{e}_1^*, \mathbf{e}_2^*$ has infinity norm bounded by b, and $\mathbf{B}^T\mathbf{s}^* + \mathbf{e}_1^* = \mathbf{c}_1 \bmod q$, $\mathbf{G}^T\mathbf{s}^* + \mathbf{e}_2^* + \lfloor q/2 \rfloor d^* = \mathbf{c}_2 \bmod q$, where $\mathbf{G} = \mathcal{H}_1(\mathsf{ovk})$. Now observe that, $(\mathbf{c}_1, \mathbf{c}_2)$ is a correct encryption of d^*, the opening algorithm $\mathsf{Open}(\mathbf{T_B}, M^*, \Sigma^*)$ must return d^*. It then follows from the requirements of the traceability game that $d^* \notin CU$. As a result, $(\mathbf{z}^*, d^*)$ is a valid forgery for Boyen's signature with respect to the verification key $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$. Furthermore, the above analysis shows that, if $\mathcal{A}$ has non-negligible success probability and runs in polynomial time, then so does $\mathcal{F}$. This concludes the proof.

5 A Ring-based Group Signature Scheme

5.1 Description of the Scheme

Let $f = x^n + 1$, where $n = 2^k$ for some $k \geq 2$, and let $N = 2^\ell = \mathsf{poly}(n)$ be the number of group users. Then we choose other scheme parameters such that ring variant of Boyen's signature scheme and the LPR encryption scheme function properly, and are secure. Let q be a prime such that $q = 1 \bmod 2n$ and $q = \mathcal{O}(\ell \cdot n^2)$. Let $\mathcal{R} = \mathbb{Z}[x]/\langle f \rangle$ and $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$. Let $m = \mathcal{O}(\log q)$. The infinity norm bound for signatures from Boyen's scheme is integer $\beta = \widetilde{\mathcal{O}}(\sqrt{\ell n})$. The norm bound for $\mathsf{Ring\text{-}LWE}$ noises is integer b such that $q/b = \ell\widetilde{\mathcal{O}}(n^{1.5})$. Choose a hash function $\mathcal{H} : \{0,1\}^* \to \{1,2,3\}^t$ to be modeled as random oracles. Let χ be a b-bounded distribution over $\mathcal{R}$.

$\mathsf{KeyGen}(1^n, 1^N)$: This algorithm performs the following steps:

1. Generate verification key $(\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u)$ and signing key $\mathbf{T_a}$ for the ring variant of Boyen's signature (see Section 2.3 for more details). Then for each $d = (d_1, \ldots, d_\ell) \in \{0,1\}^\ell$, generate $\mathsf{gsk}[d]$ as a ring-based Boyen's signature on message d.
2. Generate keys for the LPR encryption scheme: Sample $f \xleftarrow{\$} \mathcal{R}_q$ and $x, e \hookleftarrow \chi$. Then compute $g = f \otimes x + e \in \mathcal{R}_q$.

3. Output

$$\mathsf{gpk} = \big((\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u),\ (f, g)\big);\ \ \mathsf{gmsk} = x;\ \ \mathsf{gsk} = \{\mathsf{gsk}[d]\}_{d\in\{0,1\}^\ell}.$$

Sign$(\mathsf{gsk}[d], M)$: Given gpk, to sign a message $M \in \{0,1\}^*$ using the secret key $\mathsf{gsk}[d] = \mathbf{z} \in \mathcal{R}^{2m}$, the user performs the following steps:

1. Encrypt d: First extend d to $\bar{d} = (0^{n-\ell}\|d) \in \{0,1\}^n$ and view $\bar{d}$ as an element of $\mathcal{R}$ with coefficients $0-1$. Then sample $s, e_1, e_2 \hookleftarrow \chi$, and compute the ciphertext:

$$(c_1 = f \otimes s + e_1, c_2 = g \otimes s + e_2 + \lfloor q/2 \rfloor \bar{d}) \in \mathcal{R}_q^2. \tag{10}$$

2. Generate a NIZKAoK Π to show the possession of a valid message-signature pair $(d, \mathbf{z})$ for the ring variant of Boyen's signature, and that (c_1, c_2) is a correct LPR encryption of $\bar{d}$. This is done as follows:
   - Let $\mathbf{A} = \mathsf{rot}(\mathbf{a}) \in \mathbb{Z}_q^{n\times nm}$, and $\mathbf{A}_i = \mathsf{rot}(\mathbf{a}_i) \in \mathbb{Z}_q^{n\times mn}$ for every $i = 0, \ldots, \ell$. Next, consider $\mathbf{z}$ as a vector in $\mathbb{Z}^{2mn}$ with infinity norm bounded by β, and consider u as vector $\mathbf{u} \in \mathbb{Z}_q^n$. Then one has $\big[\,\mathbf{A}\,|\,\mathbf{A}_0 + \sum_{i=1}^{\ell} d_i\mathbf{A}_i\,\big]\mathbf{z} = \mathbf{u} \bmod q$.

     Furthermore, let $\mathbf{P}_0 = [\mathsf{rot}(b) \mid \mathsf{rot}(g)]^T \in \mathbb{Z}_q^{2n\times n}$ and form $\mathbf{P} = \big[\,\mathbf{P}_0\,\big|\,\mathbf{I}_{2n}\,\big] \in \mathbb{Z}_q^{2n\times 3n}$. Next, consider $\mathbf{c} = (c_1\|c_2)$ as a vector in $\mathbb{Z}_q^{2n}$, and $\mathbf{e} = (s\|e_1\|e_2)$ as a vector in $\mathbb{Z}^{3n}$. Then (10) can be equivalently written as: $\mathbf{c} = \mathbf{P}\mathbf{e} + (0^{2n-\ell} \,\|\, \lfloor q/2 \rfloor d) \bmod q$.
     The above transformation leads to the following observation:

$$\big((\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c}), d, \mathbf{z}, \mathbf{e}\big) \in \mathrm{R}_{\mathsf{gs}}(n, \ell, q, m', k_1, k_2, \beta, b),$$

     where $m' = nm$, $k_1 = 2n$, and $k_2 = 3n$.
   - Thus, the user can run the protocol for the relation $\mathrm{R}_{\mathsf{gs}}(n, \ell, q, m', k_1, k_2, \beta, b)$ in Section 3.3 with public input $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c})$ and prover's witness $(d, \mathbf{z}, \mathbf{e})$. The protocol is repeated $t = \omega(\log n)$ times to make the soundness error negligibly small, and then made non-interactive using the Fiat-Shamir heuristic as a triple $\Pi = \big(\{\mathrm{CMT}_j\}_{j=1}^t, \mathrm{CH}, \{\mathrm{RSP}_j\}_{j=1}^t\big)$, where $\mathrm{CH} = \{Ch_j\}_{j=1}^t = \mathcal{H}\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, (c_1, c_2)\big)$.
3. Output the group signature $\Sigma = \big((c_1, c_2), \Pi\big)$.

Verify$(\mathsf{gpk}, M, \Sigma)$ This deterministic algorithm works as follows:

1. Parse Σ as $\big((c_1, c_2), (\{\mathrm{CMT}_j\}_{j=1}^t, \mathrm{CH}, \{\mathrm{RSP}_j\}_{j=1}^t)\big)$.
   If $\big(Ch^{(1)}, \ldots, Ch^{(t)}\big) \neq \mathcal{H}\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, (c_1, c_2)\big)$, then return 0.
2. Then for $j = 1$ to t, run the verification step of the protocol from Section 3 with public input $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c})$ to check the validity of RSP_j with respect to CMT_j and Ch_j. If any of the conditions does not hold, then return 0.
3. Return 1.

Open$(\mathsf{gmsk}, M, \Sigma)$ On input $\mathsf{gmsk} = x$ and a signature $\Sigma = \big((c_1, c_2), \Pi\big)$, decrypt (c_1, c_2) as follows:

1. Compute $\bar{d} = \mathbf{c}_2 - x \otimes \mathbf{c}_1 \in \mathcal{R}_q$. For each $i \in [n]$, if the coefficient $\bar{d}_i$ is closer to 0 than to $\lfloor q/2 \rfloor$ modulo q, then let $\bar{d}_i = 0$; otherwise, let $\bar{d}_i = 1$.
2. If $\bar{d}$ is of the form $(0^{n-\ell}\|d)$, then return $d \in \{0,1\}^\ell$. Otherwise, return $\perp$.

5.2 Analysis

Efficiency and Correctness. The ring-based group signature scheme can be implemented in polynomial time. The group public key $\big((\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u),\ (f, g)\big)$ has bit-size $\widetilde{\mathcal{O}}(\ell n)$. In comparison with the scheme from Section 4, a factor of $\mathcal{O}(n)$ is saved. The signature size is also bounded by $\widetilde{\mathcal{O}}(\ell n)$.

The correctness of algorithm Verify follows from the facts that every group user with a valid secret key is able to compute a satisfying witness for the relation $\mathrm{R}_{\mathsf{gs}}(n, \ell, q, nm, 2n, 3n, \beta, b)\big)$, and that the underlying argument system is perfectly complete. We also set the parameters so that the LPR encryption scheme is correct, which implies that algorithm Open is also correct. The anonymity and traceability properties of the scheme are stated in Theorem 4 and 5, respectively.

Theorem 4. *In the random oracle model, the group signature scheme described in Section 5.1 is CPA-anonymous if* $\mathsf{SVP}^{\infty}_{\ell \cdot \widetilde{\mathcal{O}}(n^{3.5})}$ *on ideal lattices in the ring* $\mathcal{R}$ *is hard in the worst case.*

The proof of Theorem 4 uses the fact that the underlying argument system is statistical zero-knowledge, and the assumed hardness of the HNF variant of $\mathsf{Ring\text{-}LWE}_{n,q,\chi}$. The proof is given in Appendix C.1.

Theorem 5. *In the random oracle model, the group signature scheme described in Section 5.1 is traceable if* $\mathsf{SVP}^{\infty}_{\ell \cdot \widetilde{\mathcal{O}}(n^2)}$ *on ideal lattices in the ring* $\mathcal{R}$ *is hard in the worst case.*

The proof of Theorem 5 is similar to that of Theorem 3, and is given in Appendix C.2.

Acknowledgement. This research is supported by the Singapore Ministry of Education under Research Grant MOE2013-T2-1-041. The authors would like to thank the anonymous reviewers of PKC 2015 for their helpful comments.

References

ACJT00. Giuseppe Ateniese, Jan Camenisch, Marc Joye, and Gene Tsudik. A Practical and Provably Secure Coalition-Resistant Group Signature Scheme. In *CRYPTO*, volume 1880 of *Lecture Notes in Computer Science*, pages 255–270. Springer, 2000.

ACPS09. Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. In *CRYPTO*, volume 5677 of *Lecture Notes in Computer Science*, pages 595–618. Springer, 2009.

Ajt96. Miklós Ajtai. Generating Hard Instances of Lattice Problems (Extended Abstract). In *STOC*, pages 99–108. ACM, 1996.

AP11. Joël Alwen and Chris Peikert. Generating Shorter Bases for Hard Random Lattices. *Theory Comput. Syst.*, 48(3):535–553, 2011.

BBS04. Dan Boneh, Xavier Boyen, and Hovav Shacham. Short Group Signatures. In *CRYPTO*, volume 3152 of *Lecture Notes in Computer Science*, pages 41–55. Springer, 2004.

BCHK07. Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-Ciphertext Security from Identity-Based Encryption. *SIAM J. Comput.*, 36(5):1301–1328, 2007.

BF14. Mihir Bellare and Georg Fuchsbauer. Policy-Based Signatures. In *Public-Key Cryptography - PKC 2014 - 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings*, pages 520–537, 2014.

BMW03. Mihir Bellare, Daniele Micciancio, and Bogdan Warinschi. Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions. In *EUROCRYPT*, volume 2656 of *Lecture Notes in Computer Science*, pages 614–629. Springer, 2003.

Boy10. Xavier Boyen. Lattice Mixing and Vanishing Trapdoors: A Framework for Fully Secure Short Signatures and More. In *Public Key Cryptography*, volume 6056 of *Lecture Notes in Computer Science*, pages 499–517. Springer, 2010.

BS04. Dan Boneh and Hovav Shacham. Group Signatures with Verifier-local Revocation. In *ACM Conference on Computer and Communications Security*, pages 168–177. ACM, 2004.

CHKP10. David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai Trees, or How to Delegate a Lattice Basis. In *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 523–552. Springer, 2010.

CHL05. Jan Camenisch, Susan Hohenberger, and Anna Lysyanskaya. Compact E-Cash. In *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 302–321. Springer, 2005.

CL01. Jan Camenisch and Anna Lysyanskaya. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. In *EUROCRYPT*, volume 2045 of *Lecture Notes in Computer Science*, pages 93–118. Springer, 2001.

CNR12. Jan Camenisch, Gregory Neven, and Markus Rückert. Fully Anonymous Attribute Tokens from Lattices. In *SCN*, volume 7485 of *Lecture Notes in Computer Science*, pages 57–75. Springer, 2012.

CS97. Jan Camenisch and Markus Stadler. Efficient Group Signature Schemes for Large Groups (Extended Abstract). In *CRYPTO*, volume 1294 of *Lecture Notes in Computer Science*, pages 410–424. Springer, 1997.

CvH91. David Chaum and Eugène van Heyst. Group Signatures. In *EUROCRYPT*, volume 547 of *Lecture Notes in Computer Science*, pages 257–265. Springer, 1991.

Gen09. Craig Gentry. Fully Homomorphic Encryption Using Ideal Lattices. In *STOC*, pages 169–178. ACM, 2009.

GKV10. S. Dov Gordon, Jonathan Katz, and Vinod Vaikuntanathan. A Group Signature Scheme from Lattice Assumptions. In *ASIACRYPT 2010*, volume 6477 of *Lecture Notes in Computer Science*, pages 395–412. Springer, 2010.

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for Hard Lattices and New Cryptographic Constructions. In *STOC*, pages 197–206. ACM, 2008.

Gro04. Jens Groth. Evaluating Security of Voting Schemes in the Universal Composability Framework. In *ACNS*, volume 3089 of *Lecture Notes in Computer Science*, pages 46–60. Springer, 2004.

Gro07. Jens Groth. Fully Anonymous Group Signatures Without Random Oracles. In *ASIACRYPT*, volume 4833 of *Lecture Notes in Computer Science*, pages 164–180. Springer, 2007.

KTX08. Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Concurrently Secure Identification Schemes Based on the Worst-Case Hardness of Lattice Problems. In *ASIACRYPT*, volume 5350 of *Lecture Notes in Computer Science*, pages 372–389. Springer, 2008.

LAN02. Helger Lipmaa, N. Asokan, and Valtteri Niemi. Secure Vickrey Auctions without Threshold Trust. In *Financial Cryptography*, volume 2357 of *Lecture Notes in Computer Science*, pages 87–101. Springer, 2002.

LLLS13. Fabien Laguillaumie, Adeline Langlois, Benoît Libert, and Damien Stehlé. Lattice-Based Group Signatures with Logarithmic Signature Size. In *ASIACRYPT*, volume 8270 of *Lecture Notes in Computer Science*, pages 41–61. Springer, 2013.

LLNW14. Adeline Langlois, San Ling, Khoa Nguyen, and Huaxiong Wang. Lattice-Based Group Signature Scheme with Verifier-Local Revocation. In *Public Key Cryptography*, volume 8383 of *Lecture Notes in Computer Science*, pages 345–361. Springer, 2014.

LM06. Vadim Lyubashevsky and Daniele Micciancio. Generalized Compact Knapsacks Are Collision Resistant. In *ICALP (2)*, volume 4052 of *Lecture Notes in Computer Science*, pages 144–155. Springer, 2006.

LMPR08. Vadim Lyubashevsky, Daniele Micciancio, Chris Peikert, and Alon Rosen. SWIFFT: A Modest Proposal for FFT Hashing. In *FSE*, volume 5086 of *Lecture Notes in Computer Science*, pages 54–72. Springer, 2008.

LNSW13. San Ling, Khoa Nguyen, Damien Stehlé, and Huaxiong Wang. Improved Zero-Knowledge Proofs of Knowledge for the ISIS Problem, and Applications. In *Public Key Cryptography*, volume 7778 of *Lecture Notes in Computer Science*, pages 107–124. Springer, 2013.

LPR10. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On Ideal Lattices and Learning with Errors over Rings. In *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 1–23. Springer, 2010.

LPR13. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On Ideal Lattices and Learning with Errors over Rings. *J. ACM*, 60(6):43, 2013.

LPY12. Benoît Libert, Thomas Peters, and Moti Yung. Scalable Group Signatures with Revocation. In *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 609–627. Springer, 2012.

Lyu08. Vadim Lyubashevsky. Lattice-Based Identification Schemes Secure Under Active Attacks. In *Public Key Cryptography*, volume 4939 of *Lecture Notes in Computer Science*, pages 162–179. Springer, 2008.

MM11. Daniele Micciancio and Petros Mol. Pseudorandom Knapsacks and the Sample Complexity of LWE Search-to-Decision Reductions. In *CRYPTO*, volume 6841 of *Lecture Notes in Computer Science*, pages 465–484. Springer, 2011.

MP12. Daniele Micciancio and Chris Peikert. Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller. In *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 700–718. Springer, 2012.

MP13. Daniele Micciancio and Chris Peikert. Hardness of SIS and LWE with Small Parameters. *IACR Cryptology ePrint Archive*, 2013:69, 2013.

MV03. Daniele Micciancio and Salil P. Vadhan. Statistical Zero-Knowledge Proofs with Efficient Provers: Lattice Problems and More. In *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 282–298. Springer, 2003.

NZZ15. Phong Q. Nguyen, Jiang Zhang, and Zhenfeng Zhang. Simpler Efficient Group Signatures from Lattices. In *Public Key Cryptography*, 2015.

Pei09. Chris Peikert. Public-key Cryptosystems from the Worst-case Shortest Vector Problem: Extended Abstract. In *STOC*, pages 333–342. ACM, 2009.

PR06. Chris Peikert and Alon Rosen. Efficient Collision-Resistant Hashing from Worst-Case Assumptions on Cyclic Lattices. In *TCC*, volume 3876 of *Lecture Notes in Computer Science*, pages 145–166. Springer, 2006.

PV97. David Pointcheval and Serge Vaudenay. On Provable Security for Digital Signature Algorithms. Technical Report LIENS-96-17, Laboratoire d'Informatique de Ecole Normale Superieure, 1997.

Reg05. Oded Regev. On Lattices, Learning with Errors, Random Linear Codes, and Cryptography. In *STOC*, pages 84–93. ACM, 2005.

SSTX09. Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient Public Key Encryption Based on Ideal Lattices. In *ASIACRYPT*, volume 5912 of *Lecture Notes in Computer Science*, pages 617–635. Springer, 2009.

Ste96. Jacques Stern. A New Paradigm for Public Key Identification. *IEEE Transactions on Information Theory*, 42(6):1757–1768, 1996.

Wat05. Brent Waters. Efficient Identity-Based Encryption Without Random Oracles. In *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer, 2005.

A Security Requirements for Group Signatures

The presentation in this section follows the security model of Bellare et al. [BMW03], and the relaxed anonymity notion proposed by Boneh et al. [BBS04].

A.1 Anonymity

Consider the following anonymity experiment $\mathbf{Exp}^{\mathsf{t\text{-}anon}}_{\mathcal{GS},\mathcal{A}}(n,N)$ between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$, where $\mathsf{t} \in (\mathsf{CPA}, \mathsf{CCA})$.

Experiment $\mathbf{Exp}^{\mathsf{t\text{-}anon}}_{\mathcal{GS},\mathcal{A}}(n,N)$:

- **Initialization Phase:** The challenger $\mathcal{C}$ runs the key generation algorithm $\mathsf{KeyGen}(1^n, 1^N)$ to obtain $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk})$, then it gives $(\mathsf{gpk}, \mathsf{gsk})$ to $\mathcal{A}$.
- **Query phase 1:** If $\mathsf{t} = \mathsf{CCA}$, then $\mathcal{A}$ can make queries to the opening oracle. On input a message M and a signature Σ, the oracle returns $\mathsf{Open}(\mathsf{gmsk}, M, \Sigma)$ to $\mathcal{A}$.
- **Challenge phase:** $\mathcal{A}$ outputs two distinct identities i_0, i_1 and a message M^*. The challenger picks a coin $b \xleftarrow{\$} \{0,1\}$, computes the target signature $\Sigma^* = \mathsf{Sign}(\mathsf{gsk}[i_b], M^*)$ and sends Σ^* to $\mathcal{A}$.
- **Query phase 2:** If $\mathsf{t} = \mathsf{CCA}$, then the adversary $\mathcal{A}$ can make queries to the opening oracle. On input (M, Σ), if $(M, \Sigma) = (M^*, \Sigma^*)$, then the challenger outputs 0 and halts; otherwise it returns $\mathsf{Open}(\mathsf{gmsk}, M, \Sigma)$ to $\mathcal{A}$.
- **Guessing phase:** Finally, $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$. If $b' = b$, then $\mathcal{C}$ outputs 1, otherwise it outputs 0.

Definition 7. *Let $\mathcal{A}$ be an adversary against the anonymity of a group signature scheme $\mathcal{GS}$. Define the advantage of $\mathcal{A}$ in the above experiment as*

$$\boldsymbol{Adv}^{\mathsf{t\text{-}anon}}_{\mathcal{GS},\mathcal{A}}(n,N) = \Big|\Pr\big[\boldsymbol{Exp}^{\mathsf{t\text{-}anon}}_{\mathcal{GS},\mathcal{A}}(n,N) = 1\big] - 1/2\Big|.$$

We say that $\mathcal{GS}$ is CPA*-anonymous (respectively,* CCA*-anonymous) if for all polynomial $N(\cdot)$ and all PPT adversaries $\mathcal{A}$, the function $\boldsymbol{Adv}^{\mathsf{CPA\text{-}anon}}_{\mathcal{GS},\mathcal{A}}(n,N)$ (respectively, $\boldsymbol{Adv}^{\mathsf{CCA\text{-}anon}}_{\mathcal{GS},\mathcal{A}}(n,N)$) is negligible in the security parameter n.*

A.2 Traceability

Consider the following traceability experiment $\mathbf{Exp}^{\mathsf{trace}}_{\mathcal{GS},\mathcal{A}}(n,N)$ between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Experiment $\mathbf{Exp}^{\mathsf{trace}}_{\mathcal{GS},\mathcal{A}}(n,N)$:

- **Initialization Phase:** The challenger $\mathcal{C}$ runs $\mathsf{KeyGen}(1^n,1^N)$ to obtain $(\mathsf{gpk},\mathsf{gmsk},\mathsf{gsk})$, then it sets $CU \leftarrow \emptyset$ and gives $(\mathsf{gpk},\mathsf{gmsk})$ to $\mathcal{A}$.
- **Query Phase:** The adversary $\mathcal{A}$ can make the following queries adaptively, and in any order:
  - Secret key query: On input and index i, the challenger adds i to CU, and returns $\mathsf{gsk}[i]$ to $\mathcal{A}$.
  - Signing query: On input i, M, the challenger returns $\mathsf{Sign}(\mathsf{gsk}[i], M)$.
- **Challenge Phase:** $\mathcal{A}$ outputs a message M, and a signature Σ. The challenger proceeds as follows:
  If $\mathsf{Verify}(\mathsf{gpk}, M, \Sigma) = 0$ then return 0. If $\mathsf{Open}(\mathsf{gmsk}, M, \Sigma) = \perp$ then return 1.
  If $\exists i$ such that the following are true then return 1, else return 0:
  1. $\mathsf{Open}(\mathsf{gmsk}, M, \Sigma) = i \notin CU$,
  2. $\mathcal{A}$ has never made a signing query for i, M.

Definition 8. *Let $\mathcal{A}$ be an adversary against the traceability of a group signature scheme $\mathcal{GS}$. Define the advantage of $\mathcal{A}$ in the above experiment as*

$$\boldsymbol{Adv}^{\mathsf{trace}}_{\mathcal{GS},\mathcal{A}}(n,N) = \Pr\big[\boldsymbol{Exp}^{\mathsf{trace}}_{\mathcal{GS},\mathcal{A}}(n,N) = 1\big].$$

We say that $\mathcal{GS}$ is fully traceable if for all polynomial $N(\cdot)$ and all polynomial-time adversaries $\mathcal{A}$, the function $\boldsymbol{Adv}^{\mathsf{trace}}_{\mathcal{GS},\mathcal{A}}(n,N)$ is negligible in the security parameter n.

B Proof of Theorem 1

Let COM be a statistically hiding and computationally binding string commitment scheme. We will prove that the protocol in Figure 1 is a sZKAoK for the relation $\mathrm{R}_{\mathsf{gs}}(n,\ell,q,m,k_1,k_2,\beta,b)$; and each round of the protocol has perfect completeness, soundness error $2/3$, and communication cost $(\mathcal{O}(\ell m)\log\beta + \mathcal{O}(k_2)\log b)\log q$.

B.1 Communication Cost

As we use the commitment scheme COM from [KTX08], the commitment CMT sent by the prover P in the beginning of the interaction has bit-size $3n\log q$. The challenge Ch from the verifier V belongs to the set $\{1,2,3\}$, and thus, can be represented by 2 bits. The response RSP from P is a subset of the set of the following items:

- $2p$ permutations of $3m$ elements.
- $\bar{p}$ permutations of $3k_2$ elements.
- One permutation of 2ℓ elements.
- p vectors in $\mathbb{Z}_q^{(2\ell+2)3m}$.
- $\bar{p}$ vectors in $\mathbb{Z}_q^{3k_2}$.
- One vector in $\mathbb{Z}_q^n$.
- One vector in $\mathbb{Z}_q^{2\ell}$.

Therefore, the the bit-size of RSP is bounded by $(\mathcal{O}(\ell m)p + \mathcal{O}(k_2)\bar{p})\log q$. Recall that $p = \lfloor\log\beta\rfloor + 1$ and $\bar{p} = \lfloor\log b\rfloor + 1$, we obtain that the overall communication cost of the protocol is bounded by $(\mathcal{O}(\ell m)\log\beta + \mathcal{O}(k_2)\log b)\log q$.

B.2 Completeness

We will show that, given the public input $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c})$ if the honest prover P possesses a valid witness $(d, \mathbf{z}, \mathbf{e})$, and he follows the protocol, then he always gets accepted by V. We first recall that after the pre-interaction preparation, P obtains $d^* \in \mathsf{B}_{2\ell}$, $\mathbf{z}_1, \ldots, \mathbf{z}_p \in \mathsf{VALID}(d^*)$, and $\mathbf{e}_1, \ldots, \mathbf{e}_{\bar{p}} \in \mathsf{B}_{3k_2}$ satisfying:

$$\mathbf{A}^*\big(\sum_{j=1}^{p} \beta_j \mathbf{z}_j\big) = \mathbf{u} \bmod q \;\; \wedge \;\; \mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j \mathbf{e}_j\big) + \mathbf{Q}d^* = \mathbf{c} \bmod q, \tag{11}$$

where $\mathbf{A}^*, \mathbf{P}^*, \mathbf{Q}$ are the extended matrices formed from the public input (as in Section 3.3).

Now we will demonstrate that P passes the verification steps for every $Ch \in \{1, 2, 3\}$. Indeed, apart from the checks for correct computations, which are obviously true, it suffices to note that:

- Case $Ch = 1$: One has $\mathbf{t}_d = \tau(d^*) \in \mathsf{B}_{2\ell}$ because $d^* \in \mathsf{B}_{2\ell}$, and the set $\mathsf{B}_{2\ell}$ is invariant under permutations from $\mathsf{S}_{2\ell}$. Similarly, for all $j \in [\bar{p}]$, one has $\mathbf{t}_{\mathbf{e}}^{(j)} = \phi_j(\mathbf{e}_j) \in \mathsf{B}_{3k_2}$, as $\mathbf{e}_j \in \mathsf{B}_{3k_2}$, and the set B_{3k_2} is invariant under permutations from S_{3k_2}. Furthermore, as discussed in Section 3.1, for all $j \in [p]$, one has $\mathbf{t}_{\mathbf{z}}^{(j)} = F_{\pi_j,\psi_j,\tau}(\mathbf{z}_j) \in \mathsf{VALID}(\mathbf{t}_d)$.
- Case $Ch = 2$: The critical point is the check with respect to $\mathbf{c}_1$. The honest prover should pass this step, since, by (11) the following are true:

$$\begin{cases} \mathbf{A}^*(\sum_{j=1}^{p} \beta_j \mathbf{w}_{\mathbf{z}}^{(j)}) - \mathbf{u} = \mathbf{A}^* \sum_{j=1}^{p} \beta_j(\mathbf{z}_j + \mathbf{r}_{\mathbf{z}}^{(j)}) - \mathbf{u} = \mathbf{A}^*(\sum_{j=1}^{p} \beta_j \mathbf{r}_{\mathbf{z}}^{(j)}) \bmod q, \\ \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_j \mathbf{w}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\mathbf{w}_d - \mathbf{c} = \mathbf{P}^* \sum_{j=1}^{\bar{p}} b_j(\mathbf{e}_j + \mathbf{r}_{\mathbf{e}}^{(j)}) + \mathbf{Q}(d^* + \mathbf{r}_d) - \mathbf{c} \\ \qquad\qquad = \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{r}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\mathbf{r}_d \bmod q. \end{cases}$$

It then follows from the above discussion that the given protocol has perfect completeness.

B.3 Statistical Zero-knowledge Property

To prove that the given protocol is statistically zero-knowledge, we construct an efficient simulator $\mathcal{S}$ interacting with a (possibly cheating) verifier $\widehat{V}$, such that, given only the public input, the simulator outputs with probability negligibly close to 2/3 a simulated transcript that is statistically close to the one produced by the honest prover in the real interaction. The construction of $\mathcal{S}$ follows the standard simulation technique for Stern-type protocols ([Ste96,KTX08,LNSW13]).

The simulator $\mathcal{S}$ begins by selecting a random $\overline{Ch} \in \{1, 2, 3\}$. This is a prediction of the challenge value that $\widehat{V}$ will *not* choose.

Case $\overline{Ch} = 1$: $\mathcal{S}$ proceeds as follows:

1. Compute $\mathbf{z}'_1, \ldots, \mathbf{z}'_p \in \mathbb{Z}_q^{(2\ell+2)3m}$ such that $\mathbf{A}^* \cdot (\sum_{j=1}^{p} \cdot \beta_j \cdot \mathbf{z}'_j) = \mathbf{u} \bmod q$. This can efficiently be done using linear algebra.
2. Compute $\mathbf{e}'_1, \ldots, \mathbf{e}'_{\bar{p}} \in \mathbb{Z}_q^{3k}$; and $d' \in \mathbb{Z}_q^{2\ell}$ such that $\mathbf{P}^*\big(\sum_{j=1}^{\bar{p}} b_j \mathbf{e}'_j\big) + \mathbf{Q}d' = \mathbf{c} \bmod q$. This can also efficiently be done using basic linear algebra.
3. Now $\mathcal{S}$ samples uniformly random vectors and permutations, and sends the commitment computed in the same manner as of the real prover. Namely, it samples:

$$\begin{cases} \mathbf{r}_{\mathbf{z}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{z}}^{(p)} \xleftarrow{\$} \mathbb{Z}_q^{(2\ell+2)3m}; \; \mathbf{r}_{\mathbf{e}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{e}}^{(\bar{p})} \xleftarrow{\$} \mathbb{Z}_q^{3k}; \; \mathbf{r}_d \xleftarrow{\$} \mathbb{Z}_q^{2\ell} \\ \tau \xleftarrow{\$} \mathsf{S}_{2\ell}; \; \pi_1, \ldots, \pi_p, \psi_1, \ldots, \psi_p \xleftarrow{\$} \mathsf{S}_{3m}; \; \phi_1, \ldots, \phi_{\bar{p}} \xleftarrow{\$} \mathsf{S}_{3k}, \end{cases}$$

and sends $\mathrm{CMT} = \big(\mathbf{c}'_1, \mathbf{c}'_2, \mathbf{c}'_3\big)$ to $\widehat{V}$, where

$$\begin{cases} \mathbf{c}'_1 = \mathsf{COM}\big(\tau;\, \{\pi_j\}_{j=1}^{p};\, \{\psi_j\}_{j=1}^{p};\, \{\phi_j\}_{j=1}^{\bar{p}};\, \mathbf{A}^*(\sum_{j=1}^{p} \beta_j \mathbf{r}_{\mathbf{z}}^{(j)});\, \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{r}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\mathbf{r}_d\big), \\ \mathbf{c}'_2 = \mathsf{COM}\big(\{F_{\pi_j,\psi_j,\tau}(\mathbf{r}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\, \{\phi_j(\mathbf{r}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\, \tau(\mathbf{r}_d)\big), \\ \mathbf{c}'_3 = \mathsf{COM}\big(\{F_{\pi_j,\psi_j,\tau}(\mathbf{z}'_j + \mathbf{r}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\, \{\phi_j(\mathbf{e}'_j + \mathbf{r}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\, \tau(d' + \mathbf{r}_d)\big). \end{cases} \tag{12}$$

Receiving a challenge Ch from $\widehat{V}$, the simulator responds as follows:

- If $Ch = 1$: Output $\perp$ and abort.
- If $Ch = 2$: Send

$$\mathrm{RSP} = \big(\tau;\, \{\pi_j\}_{j=1}^{p};\, \{\psi_j\}_{j=1}^{p};\, \{\phi_j\}_{j=1}^{\bar{p}};\, \{\mathbf{z}'_j + \mathbf{r}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\, \{\mathbf{e}'_j + \mathbf{r}_{\mathbf{e}}^{(j)}\}_{j=1}^{\widehat{p}};\, d' + \mathbf{r}_d;\, \big).$$

- If $Ch = 3$: Send

$$\mathrm{RSP} = \big(\tau;\, \{\pi_j\}_{j=1}^{p};\, \{\psi_j\}_{j=1}^{p};\, \{\phi_j\}_{j=1}^{\bar{p}};\, \{\mathbf{r}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\, \{\mathbf{r}_{\mathbf{e}}^{(j)}\}_{j=1}^{\widehat{p}};\, \mathbf{r}_d\big).$$

Case $\overline{Ch} = 2$: $\mathcal{S}$ samples

$$\begin{cases} d' \xleftarrow{\$} \mathsf{B}_{2\ell};\, \mathbf{z}'_1, \ldots, \mathbf{z}'_p \xleftarrow{\$} \mathsf{VALID}(d');\, \mathbf{e}'_1, \ldots, \mathbf{e}'_{\bar{p}} \xleftarrow{\$} \mathsf{B}_{3k}; \\ \mathbf{r}_{\mathbf{z}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{z}}^{(p)} \xleftarrow{\$} \mathbb{Z}_q^{(2\ell+2)3m};\, \mathbf{r}_{\mathbf{e}}^{(1)}, \ldots, \mathbf{r}_{\mathbf{e}}^{(\bar{p})} \xleftarrow{\$} \mathbb{Z}_q^{3k};\, \mathbf{r}_d \xleftarrow{\$} \mathbb{Z}_q^{2\ell} \\ \tau \xleftarrow{\$} \mathsf{S}_{2\ell};\, \pi_1, \ldots, \pi_p, \psi_1, \ldots, \psi_p \xleftarrow{\$} \mathsf{S}_{3m};\, \phi_1, \ldots, \phi_{\bar{p}} \xleftarrow{\$} \mathsf{S}_{3k}, \end{cases}$$

and sends the commitment CMT computed in the same manner as in (12). Receiving a challenge Ch from $\widehat{V}$, it responds as follows:

- If $Ch = 1$: Send

$$\mathrm{RSP} = \big(\{F_{\pi_j,\psi_j,\tau}(\mathbf{z}'_j)\}_{j=1}^{p};\, \{F_{\pi_j,\psi_j,\tau}(\mathbf{r}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\, \{\phi_j(\mathbf{e}'_j)\}_{j=1}^{\bar{p}};\, \{\phi_j(\mathbf{r}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\, \tau(d');\, \tau(\mathbf{r}_d)\big).$$

- If $Ch = 2$: Output $\perp$ and abort.
- If $Ch = 3$: Send

$$\mathrm{RSP} = \big(\tau;\, \{\pi_j\}_{j=1}^{p};\, \{\psi_j\}_{j=1}^{p};\, \{\phi_j\}_{j=1}^{\bar{p}};\, \{\mathbf{r}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\, \{\mathbf{r}_{\mathbf{e}}^{(j)}\}_{j=1}^{\widehat{p}};\, \mathbf{r}_d\big).$$

Case $\overline{Ch} = 3$: The simulator proceeds the preparation as in the case $\overline{Ch} = 2$ above. Then it sends the commitment $\mathrm{CMT} := (\mathbf{c}'_1, \mathbf{c}'_2, \mathbf{c}'_3)$, where $\mathbf{c}'_2, \mathbf{c}'_3$ are computed as in (12), while

$$\mathbf{c}'_1 = \mathsf{COM}\big(\tau;\, \{\pi_j\}_{j=1}^{p};\, \{\psi_j\}_{j=1}^{p};\, \{\phi_j\}_{j=1}^{\bar{p}};\, \mathbf{A}^* \sum_{j=1}^{p} \beta_j(\mathbf{z}'_j + \mathbf{r}_{\mathbf{z}}^{(j)}) - \mathbf{u};\, \mathbf{P}^* \sum_{j=1}^{\bar{p}} b_i(\mathbf{z}'_j + \mathbf{r}_{\mathbf{e}}^{(j)}) + \mathbf{Q}(d' + \mathbf{r}_d) - \mathbf{c}\big).$$

Receiving a challenge Ch from $\widehat{V}$, it responds as follows:

- If $Ch = 1$: Send RSP computed as in the case $(\overline{Ch} = 2, Ch = 1)$.
- If $Ch = 2$: Send RSP computed as in the case $(\overline{Ch} = 1, Ch = 2)$.
- If $Ch = 3$: Output $\perp$ and abort.

We observe that, in every case we have considered above, since COM is statistically hiding, the distribution of the commitment CMT and the distribution of the challenge Ch from $\widehat{V}$ are statistically close to those in the real interaction. Hence, the probability that the simulator outputs $\perp$ is negligibly close to 1/3. Moreover, one can check that whenever the simulator does not halt, it will provide a successful transcript, and the distribution of the transcript is statistically close to that of the prover in the real interaction. Hence, we have constructed a simulator that can successfully impersonate the honest prover with probability 2/3.

B.4 Argument of Knowledge

We will prove that the given protocol is an AoK for the relation $\mathrm{R}(n, \ell, q, m, k_1, k_2, \beta, b)$ by showing that it satisfies the special soundness property. Namely, we will demonstrate that, for public input $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c})$, if there exists a (possibly cheating) prover $\widehat{P}$ who can correctly respond to all 3 challenges with respect to the same commitment CMT, then there exists an efficient knowledge extractor $\mathcal{K}$ who produces $(d, \mathbf{z}, \mathbf{e})$ such that:

$$\big((\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}, \mathbf{P}, \mathbf{c}), d, \mathbf{z}, \mathbf{e}\big) \in \mathrm{R}(n, \ell, q, m, k, \beta, b).$$

Indeed, based on the 3 valid responses of $\widehat{P}$, the extractor $\mathcal{K}$ can extract the following:

$$\begin{cases} \mathbf{t}_d \in \mathsf{B}_{2\ell};\ \mathbf{t}_{\mathbf{z}}^{(j)} \in \mathsf{VALID}(\mathbf{t}_d),\ \forall j \in [p];\ \mathbf{t}_{\mathbf{e}}^{(j)} \in \mathsf{B}_{3k},\ \forall j \in [\bar{p}], \\ \mathbf{c}_1 = \mathsf{COM}\big(\widehat{\tau};\ \{\widehat{\pi}_j\}_{j=1}^{p};\ \{\widehat{\psi}_j\}_{j=1}^{p};\ \{\widehat{\phi}_j\}_{j=1}^{\bar{p}};\ \mathbf{A}^*(\textstyle\sum_{j=1}^{p} \beta_j \mathbf{w}_{\mathbf{z}}^{(j)}) - \mathbf{u};\ \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{w}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\mathbf{w}_d - \mathbf{c}\big) \\ \quad = \mathsf{COM}\big(\widetilde{\tau};\ \{\widetilde{\pi}_j\}_{j=1}^{p};\ \{\widetilde{\psi}_j\}_{j=1}^{p};\ \{\widetilde{\phi}_j\}_{j=1}^{\bar{p}};\ \mathbf{A}^*(\textstyle\sum_{j=1}^{p} \beta_j \mathbf{y}_{\mathbf{z}}^{(j)});\ \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{y}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\widetilde{\mathbf{y}}_d\big), \\ \mathbf{c}_2 = \mathsf{COM}\big(\{\mathbf{v}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{v}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{v}_d\big) = \mathsf{COM}\big(\ \{F_{\widetilde{\pi}_j, \widetilde{\psi}_j, \widetilde{\tau}}(\mathbf{y}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\ \{\widetilde{\phi}_j(\mathbf{y}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\ \widetilde{\tau}(\mathbf{y}_d)\big), \\ \mathbf{c}_3 = \mathsf{COM}\big(\{\mathbf{t}_{\mathbf{z}}^{(j)} + \mathbf{v}_{\mathbf{z}}^{(j)}\}_{j=1}^{p};\ \{\mathbf{t}_{\mathbf{e}}^{(j)} + \mathbf{v}_{\mathbf{e}}^{(j)}\}_{j=1}^{\bar{p}};\ \mathbf{t}_d + \mathbf{v}_d\big) \\ \quad = \mathsf{COM}\big(\ \{F_{\widehat{\pi}_j, \widehat{\psi}_j, \widehat{\tau}}(\mathbf{w}_{\mathbf{z}}^{(j)})\}_{j=1}^{p};\ \{\widehat{\phi}_j(\mathbf{w}_{\mathbf{e}}^{(j)})\}_{j=1}^{\bar{p}};\ \widehat{\tau}(\mathbf{w}_d)\big). \end{cases}$$

Since COM is computationally binding, $\mathcal{K}$ then obtains that:

$$\begin{cases} \mathbf{t}_d \in \mathsf{B}_{2\ell};\ \widehat{\tau} = \widetilde{\tau};\ \mathbf{v}_d = \widetilde{\tau}(\mathbf{y}_d);\ \mathbf{t}_d + \mathbf{v}_d = \widehat{\tau}(\mathbf{w}_d); \\ \mathbf{A}^*(\sum_{j=1}^{p} \beta_j \mathbf{w}_{\mathbf{z}}^{(j)}) - \mathbf{u} = \mathbf{A}^*(\sum_{j=1}^{p} \beta_j \mathbf{y}_{\mathbf{z}}^{(j)}) \bmod q; \\ \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{w}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\mathbf{w}_d - \mathbf{c} = \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{y}_{\mathbf{e}}^{(j)}) + \mathbf{Q}\mathbf{y}_d \bmod q; \\ \forall j \in [p] : \widehat{\pi}_j = \widetilde{\pi}_j;\ \widehat{\psi}_j = \widetilde{\psi}_j;\ \mathbf{v}_{\mathbf{z}}^{(j)} = F_{\widetilde{\pi}_j, \widetilde{\psi}_j, \widetilde{\tau}}(\mathbf{y}_{\mathbf{z}}^{(j)});\ \mathbf{t}_{\mathbf{z}}^{(j)} + \mathbf{v}_{\mathbf{z}}^{(j)} = F_{\widehat{\pi}_j, \widehat{\psi}_j, \widehat{\tau}}(\mathbf{w}_{\mathbf{z}}^{(j)});\ \mathbf{t}_{\mathbf{z}}^{(j)} \in \mathsf{VALID}(\mathbf{t}_d); \\ \forall j \in [\bar{p}] : \widehat{\phi}_j = \widetilde{\phi}_j;\ \mathbf{v}_{\mathbf{e}}^{(j)} = \widetilde{\phi}_j(\mathbf{y}_{\mathbf{e}}^{(j)});\ \mathbf{t}_{\mathbf{e}}^{(j)} + \mathbf{v}_{\mathbf{e}}^{(j)} = \widehat{\phi}_j(\mathbf{w}_{\mathbf{e}}^{(j)});\ \mathbf{t}_{\mathbf{e}}^{(j)} \in \mathsf{B}_{3k}. \end{cases}$$

Let $d^* = \mathbf{w}_d - \mathbf{y}_d = \widehat{\tau}^{-1}(\mathbf{t}_d)$; for each $j \in [p]$, let $\mathbf{z}_j = \mathbf{w}_{\mathbf{z}}^{(j)} - \mathbf{y}_{\mathbf{z}}^{(j)} = F^{-1}_{\widehat{\pi}_j, \widehat{\psi}_j, \widehat{\tau}}(\mathbf{t}_{\mathbf{z}}^{(j)})$; and for each $j \in [\bar{p}]$, let $\mathbf{e}_j = \mathbf{w}_{\mathbf{e}}^{(j)} - \mathbf{y}_{\mathbf{e}}^{(j)} = \widehat{\phi}_j^{-1}(\mathbf{t}_{\mathbf{e}}^{(j)})$. Then it follows that $d^* \in \mathsf{B}_{2\ell}$; and $\mathbf{z}_j \in \mathsf{VALID}(\widehat{\tau}^{-1}(\mathbf{t}_d)) = \mathsf{VALID}(d^*)$, for all $j \in [p]$; and $\mathbf{e}_j \in \mathsf{B}_{3k}$ for all $j \in [\bar{p}]$. Moreover:

$$\begin{cases} \mathbf{A}^*(\sum_{j=1}^{p} \beta_j \mathbf{z}_j) = \mathbf{u} \bmod q \\ \mathbf{P}^*(\sum_{j=1}^{\bar{p}} b_i \mathbf{e}_j) + \mathbf{Q}d^* = \mathbf{c} \bmod q. \end{cases}$$

Now let $d^* = (d_1, \ldots, d_\ell, d_{\ell+1}, \ldots, d_{2\ell})$ and let $d = (d_1, \ldots, d_\ell) \in \{0,1\}^\ell$. Then $\mathcal{K}$ extracts $\mathbf{z}$ and $\mathbf{e}$ as follows:

- Let $\mathbf{z}^* = \sum_{j=1}^{p} \beta_j \mathbf{z}_j \in \mathbb{Z}^{(2\ell+2)3m}$, then it is true that $\|\mathbf{z}^*\|_\infty \le \sum_{j=1}^{p} \beta_j \|\mathbf{z}_j\|_\infty \le \beta$ and $\mathbf{A}^*\mathbf{z}^* = \mathbf{u} \bmod q$. Moreover, since $\mathbf{z}_j \in \mathsf{VALID}(d^*)$, for all $j \in [p]$, there exist $\mathbf{x}^*, \mathbf{y}^* \in \mathbb{Z}^{3m}$, whose infinity norms are bounded by β, such that $\mathbf{z}^* = \big(\mathbf{x}^* \| \mathbf{y}^* \| d_1 \mathbf{y}^* \| \ldots \| d_{2\ell} \mathbf{y}^*\big)$. Now let $\mathbf{z} = (\mathbf{x} \| \mathbf{y}) \in \mathbb{Z}^{2m}$, where $\mathbf{x}$ and $\mathbf{y}$ are obtained by dropping the last $2m$ coordinates from $\mathbf{x}^*$ and $\mathbf{y}^*$, respectively. Then one has $\|\mathbf{z}\|_\infty \le \beta$, and $\big[\mathbf{A} \big| \mathbf{A}_0 + \sum_{i=1}^{\ell} d_i \mathbf{A}_i\big]\mathbf{z} = \mathbf{u} \bmod q$.
- Similarly, let $\mathbf{e}^* = \sum_{j=1}^{\bar{p}} b_i \mathbf{e}_j$, then it is true that $\|\mathbf{e}^*\|_\infty \le \sum_{j=1}^{\bar{p}} b_j \|\mathbf{e}_j\|_\infty \le b$, and that $\mathbf{P}^*\mathbf{e}^* + \mathbf{Q}d^* = \mathbf{c} \bmod q$. Now let $\mathbf{e} \in \mathbb{Z}^k$ be the vector obtained by dropping the last $2k$ coordinates from $\mathbf{e}^*$, then $\|\mathbf{e}\|_\infty \le b$, and $\mathbf{P}\mathbf{e} + (0^{k-\ell} \| \lfloor q/2 \rfloor d) = \mathbf{c} \bmod q$

$\mathcal{K}$ finally outputs $(d, \mathbf{z}, \mathbf{e})$, which is a satisfying witness for the relation $\mathrm{R}(n, \ell, q, m, k_1, k_2, \beta, b)$. This concludes the proof.

C Security Proofs for the Ring-based Group Signature

C.1 Proof of CPA-anonymity

Let $\mathcal{A}$ be any PPT adversary attacking the CPA-anonymity of the ring-based signature scheme with advantage ϵ. Using the statistical zero-knowledge property of the underlying argument system, and the Ring-LWE assumption, we will prove that $\epsilon = \mathsf{negl}(n)$. Specifically, we construct a sequence of indistinguishable experiments $G_0^{(b)}, G_1^{(b)}, G_2^{(b)}, G_3$, such that, $\mathbf{Adv}_{\mathcal{A}}(G_0^{(b)}) = \epsilon$ and $\mathbf{Adv}_{\mathcal{A}}(G_3) = 0$.

Experiment $G_0^{(b)}$. This is the real CPA-anonymity game. The challenger runs $\mathsf{KeyGen}(1^n, 1^N)$ to obtain $(\mathsf{gpk}, \mathsf{gmsk} = x, \{\mathsf{gsk}[d]\}_{d\in\{0,1\}^\ell})$, and then gives gpk and $\{\mathsf{gsk}[d]\}_{d\in\{0,1\}^\ell}$ to $\mathcal{A}$. In the challenge phase, $\mathcal{A}$ sends a message M together with two indices $d_0, d_1 \in \{0,1\}^\ell$. The challenger sends back a challenge signature $\Sigma^* = \big((c_1^*, c_2^*), \Pi^*, \big) \leftarrow \mathsf{Sign}(\mathsf{gpk}, \mathsf{gsk}[d_b])$. The adversary then outputs $b' \in \{0,1\}$. The experiment returns 1 if $b' = b$ or 0 otherwise. We remark that, in this experiment, queries to the random oracles $\mathcal{H}$ are responded with truly uniformly random elements in $\{1,2,3\}^t$. By assumption, $\mathcal{A}$ has advantage ϵ in this experiment.

Experiment $G_1^{(b)}$. In this experiment, the following modification is introduced: Instead of faithfully generating the NIZKAoK Π^*, the challenger simulates it without using the witness. This is done by running the simulator for the underlying interactive protocol for each $j \in [t]$, and then programming the random oracle $\mathcal{H}$ accordingly. The challenge signature $\Sigma^* = \big((c_1^*, c_2^*), \Pi^*\big)$ is statistically close to the one in experiment $G_0^{(b)}$, because the argument system is statistically zero-knowledge. As a result, experiments $G_0^{(b)}$ and $G_1^{(b)}$ are indistinguishable.

Experiment $G_2^{(b)}$. In this experiment, we modify the generation of the ciphertext (c_1^*, c_2^*). Recall that in experiment $G_1^{(b)}$, one has

$$(c_1^* = f \otimes s + e_1, c_2^* = g \otimes s + e_2 + \lfloor q/2 \rfloor \bar{d}_b) \in \mathcal{R}_q^2.$$

where $f, g \in \mathcal{R}_q$ are uniformly random, and $s, e_1, e_2 \in \mathcal{R}$ are sampled from distribution χ. Now we instead let $(c_1^* = z_1, c_2^* = z_2 + \lfloor q/2 \rfloor \bar{d}_b)$, where $z_1, z_2 \xleftarrow{\$} \mathcal{R}_q$. The assumed hardness of the HNF variant of the $\mathsf{Ring\text{-}LWE}_{n,q,\chi}$ problem for the case of 2 samples implies that $G_1^{(b)}$ and $G_2^{(b)}$ are computationally indistinguishable. Indeed, if $\mathcal{A}$ can distinguish these two experiments, then it can also distinguish two Ring-LWE samples $(f, f \otimes s + e_1), (g, g \otimes s + e_2)$ from uniform samples $(f, z_1), (g, z_2)$, which violates the Ring-LWE assumption.

Experiment G_3. In this experiment we make a conceptual modification to $G_2^{(b)}$. Namely, we set $(c_1^* = z_1', c_2^* = z_2')$, where $z_1', z_2' \xleftarrow{\$} \mathcal{R}_q$. It is clear that G_3 and $G_2^{(b)}$ are statistically indistinguishable. Moreover, since G_3 is no longer dependent on the challenger's bit b, the advantage of $\mathcal{A}$ in this experiment is 0.

It follows from the above construction that the advantage of any polynomial-time adversary attacking the CPA-anonymity of our ring-based group signature is negligible. By the reduction from SVP^∞ to Ring-LWE, for the chosen parameters, the scheme is CPA-anonymous if $\mathsf{SVP}^\infty_{\ell\cdot\widetilde{\mathcal{O}}(n^{3.5})}$ on ideal lattices in the ring $\mathcal{R}$ is hard in the worst case. This concludes the proof.

C.2 Proof of Traceability

Let $\mathcal{A}$ be an PPT traceability adversary against our group signature scheme with advantage ϵ, we construct a PPT forger $\mathcal{F}$ attacking the ring variant of Boyen's signature scheme whose advantage is polynomially related to ϵ. Since the unforgeability of the ring variant of Boyen's

signature scheme can be based on the worst-case hardness of $\mathsf{SVP}^{\infty}_{\ell \cdot \widetilde{\mathcal{O}}(n^2)}$ on ideal lattices, this completes the proof.

The forger $\mathcal{F}$ is given the verification key $(\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u)$ for the ring variant of Boyen's signature scheme. It then generates a key-pair $\big((f,g), x\big)$ for the LPR encryption scheme, and begins interacting with the adversary $\mathcal{A}$ by sending $\mathsf{gpk} = (\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u, (f,g))$ and $\mathsf{gsk} = x$, the distribution of which is statistically close to that in the real attack game. Then $\mathcal{F}$ sets $CU = \emptyset$ and handles the queries from $\mathcal{A}$ as follows:

- Queries to the random oracles $\mathcal{H}$ are handled by consistently returning uniformly random values in $\{1,2,3\}^t$. Suppose that $\mathcal{A}$ makes $Q_{\mathcal{H}}$ queries to $\mathcal{H}$, then for each $\kappa \leq Q_{\mathcal{H}}$, we let r_κ denote the answer to the κ-th query.
- Queries for the secret key $\mathsf{gsk}[d]$, for any $d \in \{0,1\}^\ell$: $\mathcal{F}$ queries its own signing oracle for the ring-based Boyen signature of d, and receives in return $\mathbf{z}_{(d)} \in \mathcal{R}^{2m}$ such that $\|\mathbf{z}_{(d)}\|_\infty \leq \beta$ and $\mathbf{a}_{(d)}\mathbf{z}_{(d)} = u \bmod q$, where $\mathbf{a}_{(d)}$ is computed in the usual way. Then $\mathcal{F}$ sets $CU := CU \cup \{d\}$ and sends $\mathbf{z}_{(d)}$ to $\mathcal{A}$.
- Queries for group signatures of user d on arbitrary message M: $\mathcal{F}$ returns with a simulated signature $\Sigma = \big((\mathbf{c}_1, \mathbf{c}_2), \Pi'\big)$, where $(\mathbf{c}_1, \mathbf{c}_2)$ are faithfully generated, while the $\mathsf{NIZKAoK}$ Π' is simulated without using the legitimate secret key (as in experiment $G_1^{(b)}$ in the proof of CPA anonymity). The zero-knowledge property of the underlying argument system guarantees that Σ is indistinguishable from a legitimate group signature.

Eventually $\mathcal{A}$ outputs a message M^* and a forged group signature

$$\Sigma^* = \big((\mathbf{c}_1, \mathbf{c}_2), (\{\mathrm{CMT}_j\}_{j=1}^t, \{Ch_j\}_{j=1}^t, \{\mathrm{RSP}_j\}_{j=1}^t)\big),$$

which satisfies the requirements of the traceability game. Then $\mathcal{F}$ exploits the forgery as follows. First, one can argue that $\mathcal{A}$ must have queried $\mathcal{H}$ on input $\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$, since otherwise, the probability that $\big(Ch_1, \ldots, Ch_t\big) = \mathcal{H}\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$ is at most 3^{-t}. Therefore, with probability at least $\epsilon - 3^{-t}$, there exists certain $\kappa^* \leq Q_{\mathcal{H}}$ such that the κ^*-th oracle queries involves the tuple $\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$. Next, $\mathcal{F}$ picks κ^* as the target forking point and replays $\mathcal{A}$ many times with the same random tape and input as in the original run. In each rerun, for the first $\kappa^* - 1$ queries, $\mathcal{A}$ is given the same answers $r_1, \ldots, r_{\kappa^*-1}$ as in the initial run, but from the κ^*-th query onwards, $\mathcal{F}$ replies with fresh random values $r'_{\kappa^*}, \ldots, r'_{q_{\mathcal{H}}} \xleftarrow{\$} \{1,2,3\}^t$. The Improved Forking Lemma of Pointcheval and Vaudenay [PV97, Lemma 7] implies that, with probability larger than $1/2$, algorithm $\mathcal{F}$ can obtain a 3-fork involving the tuple $\big(M, \{\mathrm{CMT}_j\}_{j=1}^t, \mathbf{c}_1, \mathbf{c}_2\big)$ after less than $32 \cdot Q_{\mathcal{H}_2}/(\epsilon - 3^{-t})$ executions of $\mathcal{A}$. Now, let the answers of $\mathcal{F}$ with respect to the 3-fork branches be

$$r^{(1)}_{\kappa^*} = (Ch_1^{(1)}, \ldots, Ch_t^{(1)});\ r^{(2)}_{\kappa^*} = (Ch_1^{(2)}, \ldots, Ch_t^{(2)});\ r^{(3)}_{\kappa^*} = (Ch_1^{(3)}, \ldots, Ch_t^{(3)}).$$

A simple calculation shows that: $\Pr\big[\exists j \in \{1, \ldots, t\} : \{Ch_j^{(1)}, Ch_j^{(2)}, Ch_j^{(3)}\} = \{1,2,3\}\big] = 1 - (7/9)^t$. Conditioned on the existence of such j, one parses the 3 forgeries corresponding to the fork branches to obtain $\big(\mathrm{RSP}_j^{(1)}, \mathrm{RSP}_j^{(2)}, \mathrm{RSP}_j^{(3)}\big)$. They turn out to be 3 *valid* responses with respect to 3 different challenges for the same commitment CMT_j. Since COM is assumed to be computationally-binding, we can use the knowledge extractor of the underlying argument system to extract $(d^*, \mathbf{z}^*, s^*, e_1^*, e_2^*) \in \{0,1\}^\ell \times \mathcal{R}^{2m} \times \mathcal{R} \times \mathcal{R} \times \mathcal{R}$ such that:

$$\begin{cases} \|\mathbf{z}^*\|_\infty \leq \beta;\ \ \mathbf{a}_{(d^*)}\mathbf{z}^* = u \bmod q, \\ \|e_1^*\|_\infty \leq b; \|s^*\|_\infty \leq \beta;\ \ f \otimes s^* + e_1^* = c_1 \bmod q, \\ \|e_2^*\|_\infty \leq b;\ \ g \otimes s^* + e_2^* + \lfloor q/2 \rfloor d^* = c_2 \bmod q, \end{cases}$$

Now observe that, (c_1, c_2) is a correct encryption of d^*, the opening algorithm $\mathsf{Open}(x, M^*, \Sigma^*)$ must return d^*. It then follows from the requirements of the traceability game that $d^* \notin CU$. As a result, $(\mathbf{z}^*, d^*)$ is a valid forgery for the Boyen signature with respect to the verification key $(\mathbf{a}, \mathbf{a}_0, \ldots, \mathbf{a}_\ell, u)$. Furthermore, the above analysis shows that, if $\mathcal{A}$ has non-negligible success probability and runs in polynomial time, then so does $\mathcal{F}$. This concludes the proof.